Sonexion™ Administration Guide

(3.0.0)

S-2537 Rev B

Contents

1 About Sonexion Administration Guide

The *Sonexion™ Administration Guide (3.0.0) S-2537* procedures help administrators maintain Cray Sonexion storage systems release 2.1.0-3.0.0.

Scope

This publication contains procedures and reference information to support Sonexion 3000 storage systems.

> **IMPORTANT: About cscli information in this document:** This release includes available information about release 2.1 `cscli` commands. See also the *Sonexion™ CSCLI Command Reference Guide (3.0) S-9921*.

Record of Revision

Publication	Date	Description
Sonexion Administration Guide (3.0.0 Rev B) S-2537	October 2018	Updated SSA versions and fixed broken keyref.
Sonexion Administration Guide (3.0.0 Rev A) S-2537	May 2018	Updated LDAP database table schema.
Sonexion Administration Guide (3.0.0) S-2537	June 2017	Incorporates GA Software Release 3.0.0 SU006 information and cscli command information for release 2.1.0.
Sonexion 2000 Administration Guide (2.0.0 Rev C) S-2537	May 2018	Supports software release 2.0.0 for Sonexion models 900, 1600, and 2000. Updated LNet Fine-Grained Routing, and Snapshot Analyzer (SSA) features.
Sonexion™ Administrator's Guide (1.5.0) S-2537	October 2014	Supports software release 1.5.0 for Sonexion models 900, 1600, and 2000.
Sonexion™ Administrator's Guide (1.4.0 Rev A) S-2537	July 2014	Supports software release 1.4.0 for Sonexion models 900, 1600, and 2000.
Sonexion™ Administrator's Guide (1.3.1 Rev B) S-2537	March 2014	Supports software release 1.3.1 for Sonexion models 900, 1600, and 2000.

Audience

The procedures presented in this manual are for administrators of Sonexion storage systems.

Related Publications

Publication Number	Description
Sonexion™ System Snapshot Analyzer (SSA) User Guide S-2561	How to download, install, and use Cray System Snapshot Analyzer (SSA) software.
Sonexion™ CSCLI Command Reference Guide (3.0) S-9921	CSCLI reference for release 3.0.0.
Sonexion™ System Update Bundle Installation Guide 3.x.x S-2595	Installation instructions for the system update (SU) bundle 3.x.x SUxx software releases.

Typographic Conventions

`Monospace`	A `Monospace` font indicates program code, reserved words or library functions, screen output, file names, path names, and other software constructs
`Monospaced Bold`	A **`bold monospace`** font indicates commands that must be entered on a command line.
`Oblique` or *`Italics`*	An *`oblique`* or *`italics`* font indicates user-supplied values for options in the syntax definitions
Proportional Bold	A **proportional bold** font indicates a user interface control, window name, or graphical user interface button or control.
`Alt-Ctrl-f`	`Monospaced` hypenated text typically indicates a keyboard combination

Trademarks

2 Sonexion 3000 Architecture

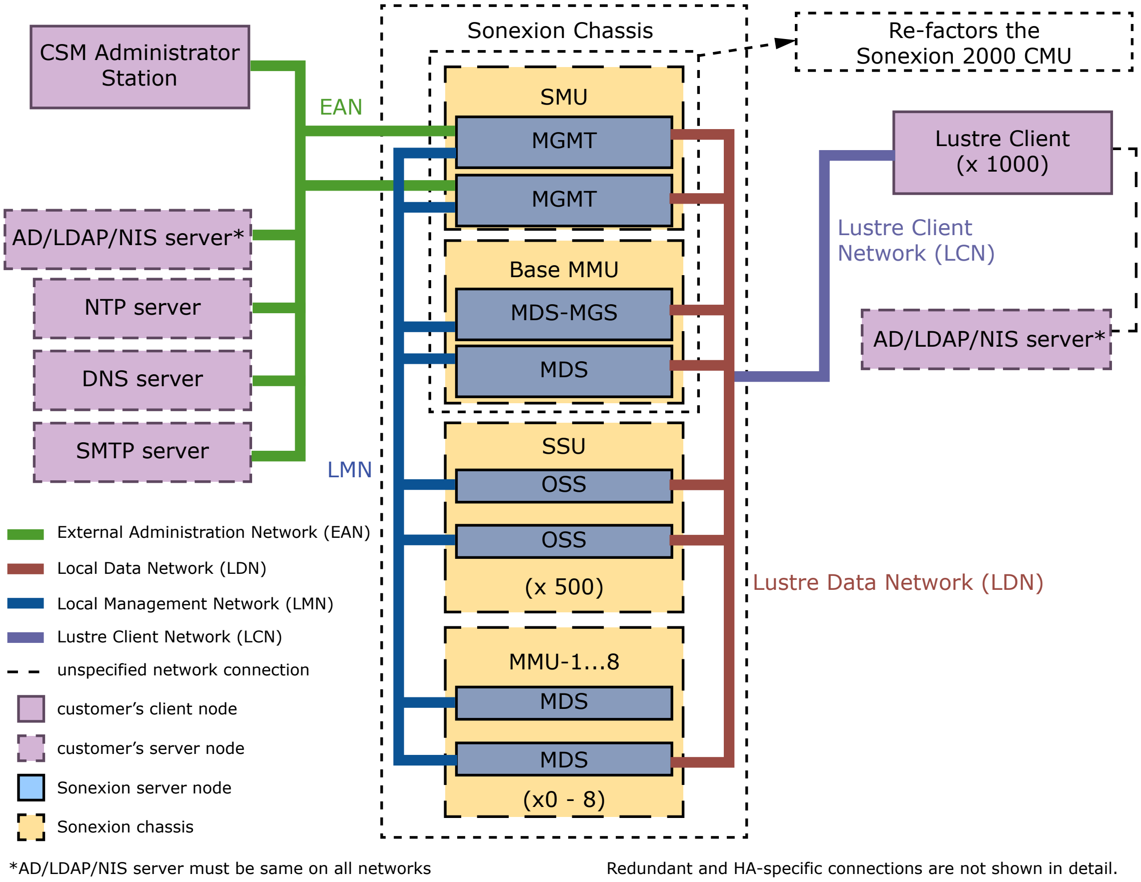
This section provides an overview of software and hardware components in the Sonexion 3000 model.

Sonexion 3000 systems consist of a unique Scale-Out Storage Architecture that consolidates Lustre® servers, RAID controllers, disk enclosures, and the operating system into an integrated storage platform that can be easily expanded using modular storage node building blocks.

Software Architecture

Sonexion 3000 software architecture consists of an integrated, multilayer software stack that includes Sonexion CSSM Manager, management (MGMT) nodes, Lustre file system, management server (MGS), metadata server (MDS), object storage server (OSS), Lustre clients, data protection layer (RAID), unified system management software (GEM), and Linux OS.

Figure 1. Sonexion 3000 Network Overview


CSM Administrator Station
EAN
AD/LDAP/NIS server*
NTP server
DNS server
SMTP server
LMN
Sonexion Chassis
SMU
MGMT
MGMT
Base MMU
MDS-MGS
MDS
SSU
OSS
OSS
(x 500)
MMU-1...8
MDS
MDS
(x0 - 8)
Re-factors the Sonexion 2000 CMU
Lustre Client (x 1000)
Lustre Client Network (LCN)
AD/LDAP/NIS server*
Lustre Data Network (LDN)
External Administration Network (EAN)
Local Data Network (LDN)
Local Management Network (LMN)
Lustre Client Network (LCN)
unspecified network connection
customer's client node
customer's server node
Sonexion server node
Sonexion chassis
*AD/LDAP/NIS server must be same on all networks
Redundant and HA-specific connections are not shown in detail.

Sonexion CSSM (Cray Sonexion System Manager) is a sophisticated management interface that provides a single-pane-of-glass view of the Sonexion 3000 infrastructure. It includes a browser-based GUI that simplifies cluster installation and configuration, and offers consolidated management and control of the storage cluster. The same operations can be executed using CSCLI. Additionally, CSSM Manager provides distributed component services to manage and monitor system hardware and software.

Management (MGMT) Nodes

Sonexion 3000 systems use primary and secondary MGMT nodes to manage the Sonexion 3000 CSSM software.

Lustre File System

The Lustre file system is a client/server based, distributed architecture that offers extreme IO performance and unparalleled scalability, making it a popular choice as a site-wide global file system in the HPC sector, serving dozens of clusters. Integrating Lustre into the Sonexion 3000 system enables users to take full advantage of the application performance of an HPC cluster environment without the traditional challenges of architectural complexity, system integration, and solution cost.

A Lustre cluster is an integrated set of servers that process metadata, and servers that store data objects and manage free space. Together, the metadata and object storage servers present file systems to clients. A Lustre cluster includes the following components:

- Lustre Management Server (MGS)
- Lustre Data Network (LDN)
- Lustre Client Network (LCN)
- MetaData Server (MDS)
- Object Storage Server (OSS)
- Lustre Clients
- Data Protection Layer (RAID)
- Unified System Management Software (GEM)
- Linux OS

Lustre Management Server (MGS)

The MGS stores configuration information for all Lustre file systems in a cluster. Each Lustre server contacts the MGS to provide information. Each Lustre client contacts the MGS to retrieve information.

MetaData Server (MDS)

The MDS (typically co-located with the MGS) makes metadata available to Lustre clients from the Metadata Target (MDT). The MDT stores file system metadata (filenames, directories, permissions, and file layouts) on disk and manages the namespace. The MDS provides network request handling for the file system.

Object Storage Server (OSS)

The OSS provides file IO service and network request handling for one or more local Object Storage Targets (OSTs). The OST stores data (files or chunks of files) on a single LUN (disk drive or drive array).

Lustre Clients

Lustre clients are nodes that use the file system to read and write data to files. Multiple clients can simultaneously read and write to different parts of the same file (distributed across multiple OSTs) maximizing the collective bandwidth of network and storage components.

Data Protection Layer (RAID)

Sonexion 3000 systems use GridRAID to provide different data protection layers throughout the solution. GridRAID adds performance to the RAID protection by replacing the Hot Spare with the Distributed Spare. The Distributed Spare is a virtual device; its space is spread across all the drives in the array. The distributed virtual nature of GridRAID allows the Repair process to write to multiple concurrent drives. In the GridRAID Repair process, data is read from all the remaining drives in the stripeset, processed and then written to the Distributed Spare.

Unified System Management Software (GEM)

Extensive Sonexion 3000 system diagnostics are managed by GEM, an advanced management software that controls and monitors Sonexion 3000s hardware infrastructure and overall system environmental conditions. GEM manages system health, provides power control to power-cycle major subsystems, monitors fans, thermals, power consumption, etc., and offers extensive event capture and logging mechanisms to support file system failover and enable post-failure analysis of hardware components.

Linux OS

Sonexion 3000 runs patched Lustre 2.7.1 software in a standard Linux environment (Scientific Linux operating system).

Hardware Architecture

Sonexion 3000 hardware architecture consists of a pre-configured, rack-level, expandable storage cluster that includes a System Management Unit (SMU), Metadata Management Unit (MMU), Scalable Storage unit (SSU), Expanded Storage Unit (ESU), network switches, and management switches. The Sonexion 3000 Management Unit (CMU) is composed of the SMU and MMU.

System Management Unit (SMU)

The SMU is made up of a 2U24 with dual Embedded Application Controllers (EACs) that provide two MGMT nodes and associated storage. There is always only one SMU in a Sonexion 3000 file system cluster, and it is always installed in the base rack. In conjunction with the base MMU, the SMU replaces the functionality of the legacy Sonexion 3000 CMU component. The SMU's principal software component is CSSM, which hosts primary and secondary MGMT nodes. The cluster interface ports support InfiniBand or 40GbE data networks and 1GbE management network connections.

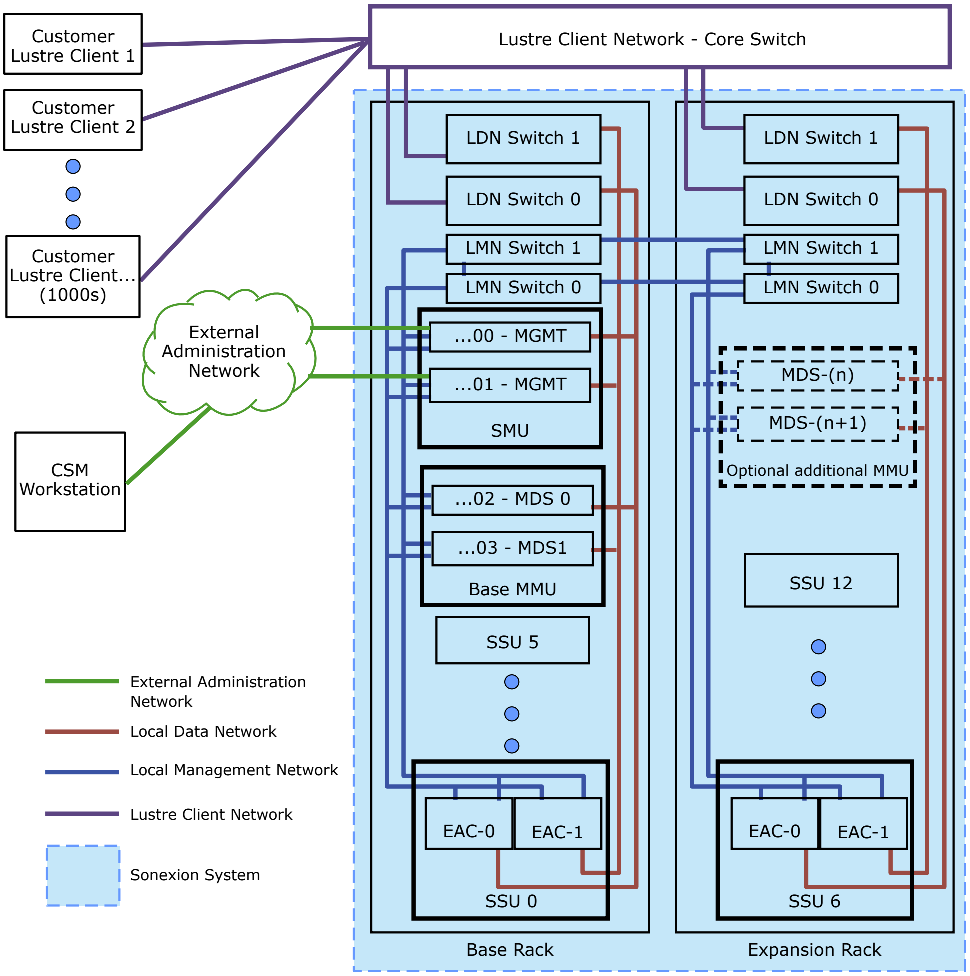
Metadata Management Unit (MMU)

The MMU is made up of a 2U24 with two EACs and associated storage that provide dual MDS nodes and dual MDTs. Lustre requires the use of the built-in DNE phase 1 feature in order to make use of multiple MDTs and multiple concurrent MDSs. MDT0 is configured as the primary MDT and also functions as the MGT. MDT1 is an additional MDT that requires the use of DNE Phase 1 to make use of it. Refer to section *Additional MMU/Additional Distributed Namespace (DNE)* for details.

Scalable Storage Unit (SSU)

Sonexion's core storage building block is the SSU. Each SSU is configured with identical hardware and software components, and hosts two OSS nodes. The SSU contains two embedded application controllers (EACs) with RAID-protected, high-availability shared storage and ports for InfiniBand or 40/100GbE data network and 1GbE management network connections.

Figure 2. SSU Scalable Storage Block Diagram



Local Data Network (LDN)

The dual InfiniBand network, with switches installed in all racks, connects all servers as needed. With the LCN, the LDN is part of the system's InfiniBand fabric.

The local data network is supported by two top-of-rack (TOR) switches per rack. Switch 0 supports even-numbered OSS servers, and switch 1 supports odd-numbered OSS servers.

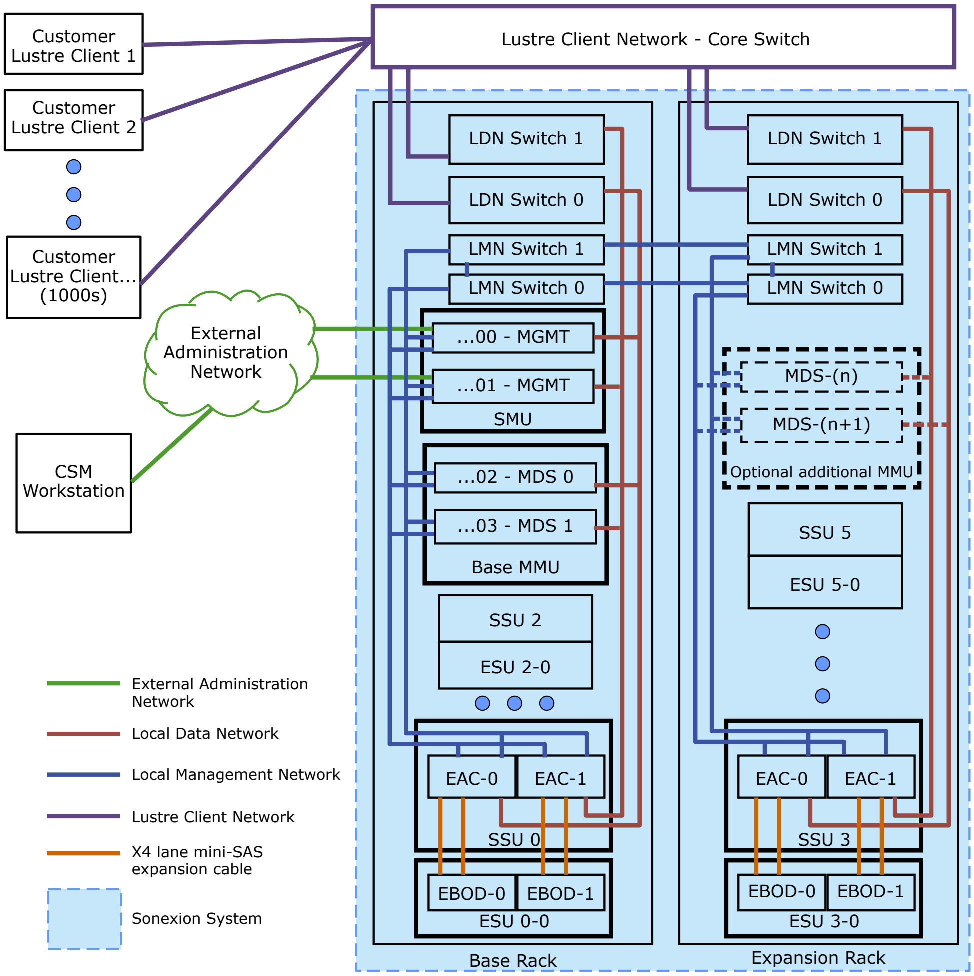
Lustre Client Network (LCN)

A high-speed data network that provides access to the Sonexion file system on the InfiniBand fabric, either via direct IB connections on the clients, or via lnet router nodes on the Cray system.

Expanded Storage Unit (ESU)

The ESU component provides the capability to expand the storage capacity of a Sonexion 3000 system by increasing the amount of storage managed by each SSU. ESUs use the same enclosure, disk drives, fans, and power supplies as SSUs but use lower cost EBOD controllers instead of EACs. The EBOD controllers are cabled to the EACs of the associated SSU that manages that ESU's storage. This increases the amount of storage on the system without increasing the number of more costly EACs. Sonexion 3000 systems are available with SSUs only or with SSU plus one ESU (SSU+1) for additional storage.

Figure 3. ESU (SSU+1) Expanded Storage Block Diagram



Lustre Network Fabric Switches

The Lustre Network Fabric Switches (InfiniBand or 40GbE) manage IO traffic and provide network redundancy throughout the Sonexion system. To maximize network reliability, dual network switches connect to the EACs in each SSU. If one switch fails, the second EAC (connected to the active switch) manages the OSTs of the EAC connected to the failed switch. The network switches are fully redundant at every point and are interconnected to provide local access from the MGMT, MGS, and MDS nodes to all storage nodes. These switches support the LDN and LCN.

Management Switches

The Management Switches (in a dual-switch configuration) provide dedicated and fully redundant dual local networks on 1GbE switches used to manage configuration and monitor health of all Sonexion 3000 components. The management networks are private and not used for data IO in the cluster.

The networks also are used for IPMI traffic to the SSU EACs, enabling them to be power-cycled by CSSM.

Additional MMU/Additional Distributed Namespace (DNE)

Lustre 2.5 supports Phase 1 of the Lustre DNE feature, which allows multiple MDTs, operating through multiple MDS nodes, to be configured and to operate as part of a single file system. This feature allows the number of metadata operations per second within a cluster to scale beyond the capabilities of a standard Sonexion 3000 system's single MDS. To achieve this capability requires that the file system namespace be configured manually so that file system operations are evenly distributed across the MDS/MDT resources.

For the Sonexion 3000, the term ADU is deprecated but may be considered synonymous with the term “additional MMU.”

Up to eight additional MMUs can be added to the file system cluster in order to increase the metadata performance and capacity of the file system. Note that the Lustre DNE Phase 1 feature must be used in order to take advantage of the additional MDTs. Together with the base MMU, a Sonexion 3000 file system supports up to a total of 9 MMUs per file system, providing up to a total of 18 MDS/MDT pairs.

3 Components and Hardware Supported by this Release

Sonexion Release 3.0

Sonexion 3000 is a next-generation HPC storage platform that delivers industry-leading performance and durability using 6/12GB architecture and the Intel Grantley/Haswell platform. The Sonexion 3000 platform builds upon Sonexion's history of HPC excellence by offering substantial upgrades and enhancements to system components and hardware. The 3.0 release currently supports only the Sonexion 3000 platform. Other platforms may be supported in future versions of 3.0, at which time the additional platform support will be announced.

CMU

The re-engineered CMU consists of two sub-components: the System Management Unit (SMU) and Metadata Management Unit (MMU), housed in separate 2U24 enclosures, which replace the Intel quad server and adjacent EBOD from the previous models.

> **NOTE:** The Advanced Metadata Management Unit (AMMU) is an optional replacement for the MMU (in the base rack), which may be used when greater single directory MDS performance is required.

SMU -- dual MGMT nodes in an HA pair:

- 2U24 enclosure
  - Dual PSUs
- Dual basic EACs
- 12 drives
  - 7 x Thunderbolt 10K HDDs (900 GB 2.5-inch)
  - 5 x Valkyrie 15K HDDs (300 GB 2.5-inch)

MMU -- dual MDS nodes in an HA pair:

- 2U24 enclosure
  - Dual PCMs
- Dual standard EACs
- 22 drives
  - 22 x Thunderbolt 10K HDDs (900 GB 2.5-inch)

AMMU – optional replacement for the MMU in the base rack:

- 2x 1U servers
  - Dual power supply modules
  - Single 1TB enterprise SATA drive

  - Dual SAS connections to the 2U24 EBOD
- 2U24 EBOD storage unit
  - 22 Drives (Thunderbolt 10K HDDs, 1800GB, 2.5-inch)

SSU – Each SSU hosts dual OSS nodes in an HA pair:

- 5U84 G2 enclosure
  - Dual PSUs
- Dual standard EACs
- 84 drives
  - 82 x HDDs (SAS 3.5-inch)
  - 2 x SSDs (SAS 2.5 inch)

ESU – 5U84 G2 enclosure with 6Gb EBOD controllers and high-capacity HDDs:

- 5U84 G2 enclosure
  - Dual PSUs
- 6Gb EBOD controllers
- 82 HDDs (SAS 3.5 inch)

Management Switches – Dual Brocade ICX6610 switches used for the management network (LMN):

- Base rack: Dual Brocade ICX6610 switches (24-port or 48-port, 1 GbE)
- Expansion rack: Dual Brocade ICX6610 switches (24-port, 1 GbE)

Network Switches –

Dual Mellanox SB7790 EDR switches are used for the Lustre client network (LCN):

- Dual Mellanox SB7790 EDR (36-port, 100Gb InfiniBand)

5U84 G2

The re-engineered 5U84 enclosure (5U84 G2) offers the following features:

- Enhanced LED display
- Improved drawer release
- Redesigned side card cover
- Improved sensor placement

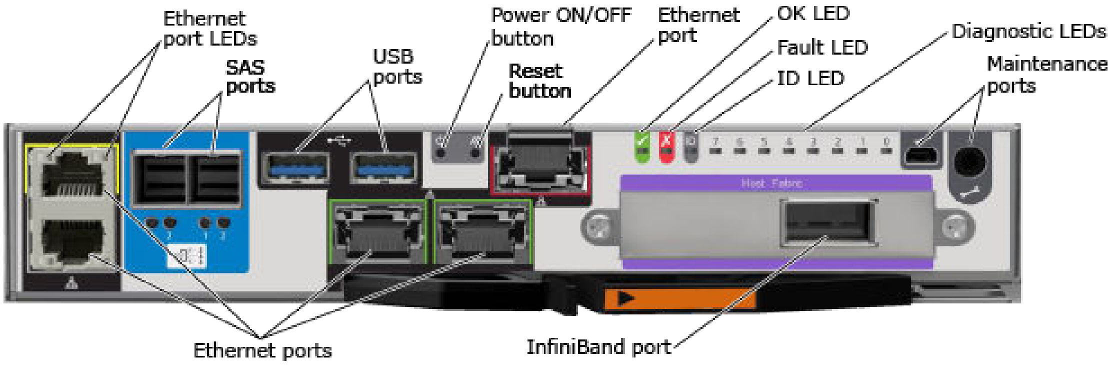
EACs

Two EACs, basic and standard, are supported:

- Basic EAC (used in SMU only)
  - 64 GB DRAM
  - E5-2609 v3 CPU
  - 12Gb SAS controller
  - Dual 128GB SSDs
  - Single-port FDR IB Local Data Network

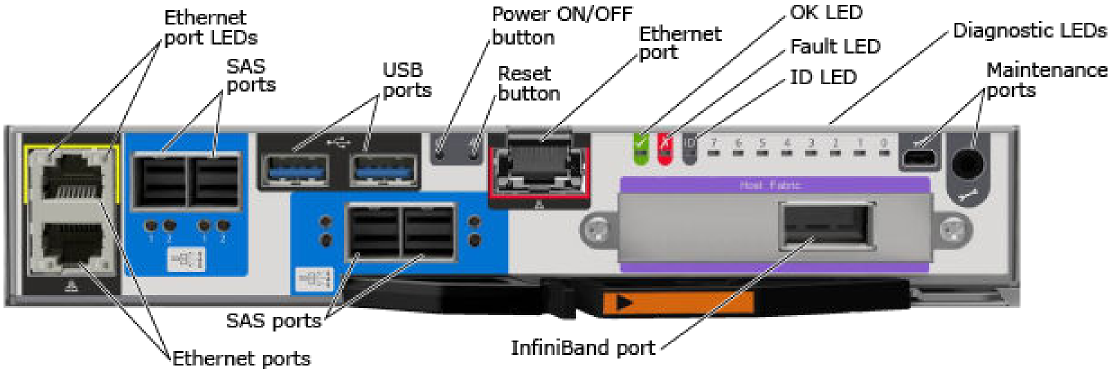
  - Mezzanine Daughter Card: Dual 1/10GbE; auxiliary ports for customer-specific administration configurations

Figure 4. EAC for SMU Nodes (Basic EAC)



- Standard EAC (used in MMUs and SSUs)
  - 64 GB DRAM
  - E5-2618L v3 CPU
  - 12Gb SAS controller
  - Single 128GB SSD
  - Single-port EDR IB Local Data Network
  - Mezzanine Daughter Card: 12 Gb SAS controller for additional bandwidth

Figure 5. EAC for MMU and SSU (Standard EAC)



For detailed descriptions of the LEDs and their functions, please see the latest revision of *Sonexion 3000 Quick Start Guide*.

4 Functionality Supported in Each Sonexion Release

This section compares features and capabilities included in the Sonexion releases covered in this manual.

4.1 Qualified Functionality in Sonexion 3.0.0

- Installation and deployment of Sonexion 3000 and 3000t systems with SSU configuration and ESUs
- High Availability:
  - SSU Node Failover/Failback
  - SMU Node Failover/Failback
  - Standard 2U MMUs or optional 4U AMMU Node Failover/Failback
  - Dual Management Network Switch Redundancy (DMN)
- Lustre 2.7, based on Intel Enterprise Edition Lustre (IEEL) 2.7
- InfiniBand
- OS: CentOS Linux 7.2
- CSCLI - Cray Sonexion Command Line Interface
- CSSM - Cray Sonexion System Manager 3.0.0
- Supported browsers: Chrome, FireFox, Safari, and Internet Explorer 11, for Windows, Linux, and MacOS
- Support File Bundle Collection
- Split SMU/MMU (or optional AMMU) architecture that combines to make a CMU
- Optional Advanced Metadata Management Unit (AMMU) as a replacement for the base MMU
- Support for GOBI OneStor USM STX_GOBI_R1.29a
- Mellanox IB EDR capabilities
- Mellanox SB7790 EDR Switches
- High-Speed Local Data Network (LDN) Interface:
  - Mellanox CX-4 - EDR / FDR HCAs
  - Mellanox CX-4 IB EDR/100GbE HCAs
- Next generation Embedded Application Controller (EAC) platform using Intel Haswell and Broadwell CPUs
- RAID Stack: Updates to optimize GridRAID and SCSI performance
- Support for 4K Native and SED Hard Drives
- Support for 10K RPM HPC Drive
- Support for 10TB Drives

- Drives: All 4K Native, T10-PI Format Type2, SED, Non-SED, and FIPS supported

Drive Model	SSD / RPM	Capacity
Seagate ST900MM0038	10K	900GB
Seagate ST300MP0075	15K	300GB
Seagate ST10000NM0016, ST10000NM0086, ST10000NM0096, ST10000NM0216, and ST10000NM0226	7K	10TB
Seagate ST4000NM0031	10K	4TB
Seagate ST4000NM0074	7K	4TB
Seagate ST6000NM0074	7K	6TB
Seagate ST8000NM0095 and ST8000NM0125	7K	8TB
Seagate ST800FM0053 and ST800FM0043	SSD	800GB
Seagate ST800FM0183	SSD	800GB

- RAS features:
  - RAS Infrastructure: CLI, Nagios and Ganglia plugins, REST API
  - Guided Walkthrough Repairs: 2U24 / 4U24 / 5U84 Drives, 2U24 / 4U24 PCMs, 5U84 cooling module
  - Fault Isolation: 5U84 cooling module, 2U24 / 5U84 I/O controllers
- Standard MMU Addition Procedure

NOT Supported in Sonexion 3.0.0

- CNG
- LMT (Lustre monitoring tool); replaced by CSSM 3.0.0.

4.2 Qualified Functionality in Sonexion 2.1.0

- Installation and deployment of Sonexion 3000 and 3000t systems with the following SSU configurations:
  - Single or Multi-SSU (SSU Only)
  - SSU + Single ESU (SSU+1)
- High Availability:
  - SSU Node Failover/Failback
  - SMU Node Failover/Failback
  - MMU Node Failover/Failback
  - Dual Management Network Switch Redundancy (DMN)
  - Dual PDU Redundancy
- Lustre 2.5.1: Lustre Performance Monitoring of LMT (ltop/lstat)
- Kernel/OS: Scientific Linux (SL) 6.5 / OS 6.2
- CSCLI - Sonexion Command-Line Interface

- CSSM - Cray Sonexion System Manager
- Chrome, Firefox, Safari, and Internet Explorer 11 browsers for Windows, Linux, and MacOS
- Support File Bundle Collection
- Split SMU/MMU architecture that combines to make a CMU
- Support for GOBI OneStor USM STX_GOBI_R1.29
- Mellanox InfiniBand QDR/FDR/EDR
- Mellanox SB7700 EDR Switches
- 40GbE for local data network (LDN)
- High Speed LDN Interface: Mellanox CX-4 - EDR / FDR HCA
- Next generation Embedded Application Controller (EAC) platform (CS-L300 and CS-L300N EACs) using Intel Haswell and Broadwell CPUs
- GridRAID: All custom GridRAID Optimal Performance and SCSI code updates and patches
- Support for 4K Native and SED Hard Drives
- Support for 10K RPM HPC Drives
- Drives: All 4K Native, T10-PI Format Type2, SED and Non-SED supported

Drive Model	SSD/RPM	Capacity	Type
Seagate ST900MM0038	10K	900GB	SED
Seagate ST300MP0075	15K	300GB	SED
Seagate ST4000NM0031	10K	4TB	SED
Seagate ST4000NM0074	7K	4TB	SED
Seagate ST6000NM0074	7K	6TB	SED
Seagate ST8000NM0095	7K	8TB	SED
Seagate ST8000NM0125	7K	8TB	SED, FIPS
Seagate ST800FM0053	SSD	800GB	SED
Seagate ST800FM0043	SSD	800GB	non-SED

- FRU replacements
- RAS (Reliability, Availability, Serviceability):
  - RAS Infrastructure: CLI, Nagios and Ganglia plugins, REST API
  - Guided Walkthrough Repairs: 2U24 / 4U24 / 5U84 Drives, 2U24 / 4U24 PCMs, 2U Quad Server PSUs
  - Fault Isolation: 5U84 cooling module, 2U24 / 5U84 I/O controllers
  - Remote Support: service events, IEMs (Interesting Event Messages), inventory snapshots
- Extra MMU (ADU/DNE) Addition Procedure
- SSU Addition Procedure (SSU Only and SSU+1)

New Functionality in Sonexion 2.1.0

- Support for GOBI OneStor USM STX_GOBI_R1.29

NOT Supported in Sonexion 2.1.0

- NXD I/O acceleration functionality
- Advanced MMU (AMMU)
- CS9000, CS6000, CS1500 platforms. These older hardware platforms continue to be supported under the ClusterStor 2.0.0 release stream.
- CIFS/NFS Gateway (CNG)
- Intel Omni-Path Data Fabric

5 Software Versions and Requirements

This section provides information about the environment and software required for the Sonexion software releases, beginning with 3.0.

CSSM Version – release 3.0, SU006

Cray Sonexion Manager 3.0.0 Build v3.0.0-98.62ec786f, 2017-04-26

SMU/MMU/SSU: GOBI OneStor USM STX_GOBI_R1.24a

ESU (5U84 Gen 2): USM r4.1.16

ESU (5U84 Gen 3): USM v5 r1.4a

AMMU EBOD: USM v5 r1.4a

Lustre Server (x86_64 Architecture) – release 3.0, SU006

Operating System

CentOS Linux 7.2

Kernel

2.6.32-431.17.1.x2.1.32.x86_64

File System

lustre-2.7.19.8.x8-43_3.10.0_327.36.3.x3.0.15.x86_64_g4f2e6fb

Required User-Supplied Network Infrastructure – release 3.0, SU006

Manual workarounds may be available for environments without these servers. Contact your support representative for more information.

DHCP Server

Provides the MGMT nodes' IP addresses for browser connections (user can choose to use a static IP address configuration for the "public" interfaces on the MGMT nodes)

NTP Server

Synchronizes clocks across the cluster's nodes

DNS Server

Resolves LDAP and NTP hostnames on the MGMT nodes

6 Sonexion Terms, Abbreviations, and Definitions

This glossary of key Sonexion terms is current to release 3.0.

5U84 G2 / 5U84 G3 (new for 3.0 SU006)

The second and third generation versions of the standard 5U84 enclosure, used for Sonexion SSUs and ESUs. The 5U84 G2 chassis has several hardware changes, including an improved LED display and drawer release, redesigned side card cover, reduced number of sensors, and a different fan module latch color. The 5U84 G3 enclosure increases the amount of backend SAS bandwidth to ~12 GB/sec vs. the 5U84 G2, which has a backend SAS bandwidth of ~8.4 GB/sec.

ADU, Additional DNE Unit

Deprecated term that refers to the additional MDS nodes and MDT storage supported by the Lustre DNE (Distributed Namespace) Phase 1 feature. For the 3000, this term is synonymous with the additional MMUs that may be optionally installed in up to 8 storage racks, 1 per rack.

AMMU, Advanced Metadata Management Unit (new for 3.0 SU006)

2 Intel 1U servers that replace the 2U24 MMU used in Sonexion 3000 2.1.0.

Base MMU, Base Metadata Management Unit

The base MMU is the MMU that is always installed in the base rack. The base MMU provides two MDS nodes along with two MDTs. MDT0 functions as the default MDT and as the root MDT for DNE phase 1. MDT1 requires DNE in order to be utilized.

Base Rack

The first rack in a Sonexion storage cluster that contains the SMU and base MMU along with the rack networking infrastructure and from 1 to 6 SSUs.

CLI, Command Line Interface

A text-based interface, used to operate software and operating systems.

CLP, Sonexion Linux Platform

Base OS used by all the rack components.

CMU, Cluster Management Unit

The Sonexion component that provides the physical deployment of the MDS, MGS, and MGMT server nodes and associated storage. This term is deprecated for the Sonexion 3000 platform and is functionally equivalent to the combination of SMU and base MMU installed in the base rack.

CIFS/NFS Gateway (new for 3.0, SU006)

2U4N server configured to export the Lustre file system to CIFS2 and NFS clients.

Critical, Critical Array State

The state of a GridRAID or MDRAID array where the subsequent failure of one more storage component may lead to the data becoming inaccessible.

CSI, Cray Sonexion Installer

Sonexion software used for manufacturing and installing Sonexion systems.

CSSM, Cray Sonexion System Manager

Sonexion platform, software and hardware management system.

CSMS, Cray Sonexion Management Server

Sonexion MGMT node; the primary and secondary instances of the CSSM software and all associated components and services running on a server node in the CMU.

CMU Storage

Sonexion 2000 storage enclosure dedicated to the CMU; deprecated for the Sonexion 3000, as the SMU and MMU have their own storage resources.

CTU, Customer Test Unit

CTU features are enabled in a release for customer evaluation, early access, and acceptance testing, but are not supported for production systems.

Data Block

A component of a “parity group” (or “stripe”) containing actual user data, also referred to as a “data chunk” or “data unit.”

Degraded, Degraded Array State

The state of a GridRAID or MDRAID array operating with one failed storage component.

Distributed Spare, Distributed Spare Volume

The aggregate collection of distributed spare data blocks in a GridRAID array that comprises a single logical spare volume for the specific GridRAID array that contains it. Each distributed spare contains the equivalent of one physical drive’s worth of distributed spare space and is used as the target of the GridRAID reconstruction process and the primary data source for the GridRAID rebalance process.

DMN, Dual LMN

Refers to the “Dual Local Management Networks” (or “Dual Management Networks”) feature supported beginning with the Sonexion 1.5.0 release.

DNE, Distributed Namespace

Lustre DNE Phase 1 feature supported in Lustre 2.5 that allows multiple MDS/MDT components to operate within a single file system.

EAC, Embedded Application Controller

SBB form factor x86 base application controller provides the CPU platform for code executing as part of the Sonexion file system cluster components.

EAN, External Administration Network

Customer administration network, external to the Sonexion system. Connected to the CSMS nodes in order to provide access to the CSSM software.

ECN, Enterprise Client Network

Refers to the 10GbE or 40GbE data network connecting non-Lustre enterprise clients to the optional CIFS NFS Gateway (CNG).

ESM, Embedded Server Module

Deprecated term for an Embedded Application Controller (EAC) because it implies general server functionality that is not supported on the dedicated Sonexion Embedded Application Controllers (EAC).

ESU, Expansion Storage Unit

A 5U84 storage enclosure with two SAS EBOD controllers installed in place of the EACs.

Expansion Rack

The additional racks (to the base rack) in a Sonexion storage cluster that contain the rack networking infrastructure and some number of SSUs. Sometimes called "storage rack."

Failed, Failed Array State

The state of a GridRAID or MDRAID array that has been failed by the system.

GB/sec, GigaBytes per Second

10^9 Bytes per second

Gbit/sec, Gigabit per Second

10^9 bits per second

GbE, Gigabit Ethernet

Ethernet standard that transmits at 1 gigabit per second.

GOBI, Generic Enclosure Management On BMC Interface (new for Sonexion 3000)

Firmware architecture for Sonexion enclosures.

GridRAID

Sonexion implementation of parity declustered RAID. A RAID level organization that combines RAID 6 data protection with a declustering methodology. GridRAID overcomes single drive throughput bottlenecks by distributing parity groups and spare space across all storage components in an array.

ICL, Inter-Controller Link

A link that connects two controllers or two servers together. Used in Sonexion as a dedicated HA communication path.

IEEL, Intel Enterprise Edition Lustre

Intel’s enterprise version of the Lustre file system.

ISL, Inter-Switch Link

A connection between two related switches.

KiB, Kibibyte

1024 bytes

LCN, Lustre Client Network

High speed data network connecting Lustre clients to the Sonexion Local Data Switches (LDS).

ldiskfs, Lustre Disk File System

Lustre version of a patched Ext4 file system.

LDN, Local Data Network

A dual InfiniBand or 40GbE network with switches installed in all racks, connecting all servers and enclosures as needed and used as uplink points to the end user client infrastructure.

LDS, Local Data Switch

A dual InfiniBand or 10GbE network switch installed in a Sonexion rack as part of the LDN and used for providing high speed data connectivity. Used as uplink points to the end user client infrastructure.

LMN, Local Management Network

A private 1GbE network connecting all Sonexion servers and enclosures.

LMS, Local Management Switch

A 1GbE switch installed in a Sonexion rack as part of the LMN and used for providing private management network connectivity for all Sonexion servers and enclosures.

LMT, Lustre Monitoring Tool

The Lustre Monitoring Tool is a tool for capturing and displaying Lustre file system activity. In this release, new Lustre performance functionality is available.

Lustre®

Open source clustered file system.

Lustre Servers

The set of Lustre servers that comprise the Lustre file system; includes the MGS, MDS, and multiple OSSes.

MDS, Metadata Server

Lustre server component that manages the Lustre file system metadata.

MDT, Metadata Target

Lustre component, a storage volume that holds the Lustre file system metadata.

MGMT, Management Server Node

One of two Sonexion management servers that provide management functions for the storage cluster.

MGMT0

The primary Sonexion management server, typically used for web access and SSH logins for managing the storage cluster.

MGMT1

The secondary management server, typically used to provide boot services to nodes in the storage cluster.

MGS, Management Server

Lustre server component that manages the Lustre MGT.

MGT, Management Target

Lustre component, the storage volume holding the Lustre file system management data that allows clients to discover, mount, and operate the file system.

MMU, Metadata Management Unit

2U24 with two EACs and associated storage provides dual MDS nodes and dual MDTs. Lustre requires the use of the built-in DNE phase 1 feature in order to make use of multiple MDTs and multiple concurrent MDSes.

NIS, Network Information Service

Maintains and distributes a central directory of user and group information in a network.

Normal, Normal Array Activity

Characterizes the activity of a GridRAID or MDRAID array that is engaged in processing I/O only and is not conducting any recovery, sync, or RAID checking activities.

NXD, NytroXD I/O Acceleration Feature

Feature that uses NytroXD flash acceleration software to cache I/Os that match specific criteria. This function will be supported in a future software release.

Offline, Array Is Offline

The array is not available.

Optimal, Optimal Array State

The state of a GridRAID or MDRAID array where all drives in the array are operational without the involvement of spare volumes or dedicated hot spares. For GridRAID this is equivalent to the “Redundant 0/2” terminology.

OSS, Object Storage Server

Lustre server component that operates and manages the Lustre OSTs.

OST, Object Storage Target

Lustre component, a storage volume that holds Lustre file system data.

Parity Block

Component of a parity group that contains protection information for the group derived from the set of data blocks in the parity group. Also referred to as a “parity chunk” or “parity unit.”

Parity Group

The set of “data blocks” and derivative “parity blocks” that together comprise a protected data set. Also referred to as a “stripe.”

PDU

Power Distribution Unit

RAID Check, RAID Consistency Check

In this process, the system periodically checks that the parity information is consistent for every “parity group” (stripe) in the array. This process is sometimes referred to as “parity scrubbing.”

RAS (Reliability, Availability, Serviceability) System

Sonexion feature providing system RAS diagnostic and repair features.

Rebalance, Rebalance Process

Phase 2 of the 2-phase GridRAID recovery process: a GridRAID array essentially copies reconstructed data from a distributed spare volume in the array to a physical replacement drive, freeing the distributed spare volume when complete for future reuse.

Rebalancing, Rebalancing Array Activity

Characterizes the activity of a GridRAID array that is engaged in the rebalance phase of the recovery process.

Reconstructing, Reconstructing Array Activity

Characterizes the activity of a GridRAID array that is engaged in the reconstruction phase of the recovery process.

Reconstruction, Reconstruction Process

Phase 1 of the 2-phase GridRAID recovery process: a GridRAID array reconstructs the data from a missing storage component onto one of the distributed spare volumes.

Recovering, Recovering Array Activity

Characterizes the activity of a GridRAID or MDRAID array that is engaged in the recovery process.

Recovery, Recovery Process

In this process, a GridRAID or MDRAID array recovers from a storage component failure.

Rebuild, Rebuild Process

In this single phase recovery process, an MDRAID array reconstructs data for a failed drive and copies it to a dedicated replacement drive.

SED, Self-Encrypted Drive

A disk drive that automatically encrypts/decrypts data to/from the media.

SMU, System Management Unit

2U24 with dual EACs that provides two MGMT nodes and associated storage. There is always only one SMU in a Sonexion file system cluster, and it is always installed in the base rack. In conjunction with the base MMU, the SMU replaces the functionality of the earlier Sonexion CMU component.

Spare Volume, GridRAID Spare Volume or Distributed Spare Volume

The aggregation of the equivalent of one drive's worth of distributed spare space considered collectively as a logical spare drive or volume and used as the target of the GridRAID repair operation.

SSU, Scalable Storage Unit

5U84 storage enclosure and dual EACs (Embedded Application Controllers), provides dual OSSes and associated storage.

SSU Addition

Refers to the process of increasing the storage capacity of a Sonexion file system by incorporating additional SSUs into the cluster.

SSU Expansion

Refers to the attachment of an ESU to each SSU, thus increasing the amount of storage managed by each SSU.

Storage Component

Refers to an individual drive when considered as part of a configured GridRAID or MDRAID array.

Storage Rack

See "Expansion Rack."

Stripe

See "Parity Group."

TB, Terabyte

10^9 bytes

7 Log In to CSSM

Use either of the following procedures to begin using the CSM, the primary administrative interface for the Sonexion system.

Log In to CSM via the CSCLI

To use the CSCLI interface, SSH into the active MGMT node:

```
[Client]$ ssh -l admin MGMT_node
```

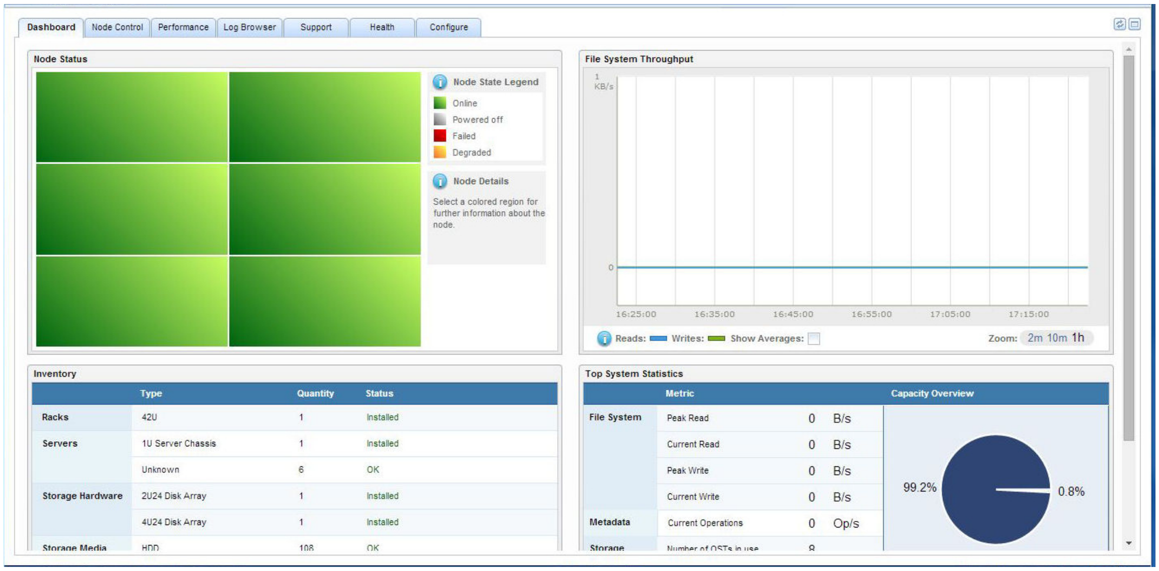
Log In to CSSM via the GUI

Access the CSM GUI using a direct connection between the site's IP network and the Sonexion MGMT nodes. Log in to the GUI using a browser on a PC or on a terminal that is connected to the site's IP network. CSSM supports Chrome, Firefox, or Internet Explorer browsers.

1. Open a browser window.
2. Enter the CSSM IP address in the browser's URL field, and press **Enter**.
3. Enter the administrator password used to access CSSM at the site, and press **Enter**.

   A **What's New** window will appear. Close this window to see the **Dashboard** tab open in the browser window.

   Figure 6. Dashboard Tab After Login



Select a colored region for further information about the node.

Inventory

	Type	Quantity	Status
Racks	42U	1	Installed
Servers	1U Server Chassis	1	Installed
	Unknown	6	OK
Storage Hardware	2U24 Disk Array	1	Installed
	4U24 Disk Array	1	Installed
Storage Media	HDD	108	OK

Top System Statistics

	Metric			Capacity Overview
File System	Peak Read	0	B/s	
	Current Read	0	B/s	
	Peak Write	0	B/s	
	Current Write	0	B/s	
Metadata	Current Operations	0	Op/s	
Storage	Number of OSTs in use	8		

The **Dashboard** tab displays individual widgets to view status and performance data. From the Dashboard, users can survey high-level system health and resolve issues with individual nodes and take further action by quickly linking to the **Node Control** tab.

8 Introduction to RAS - Reliability, Availability, and Serviceability

This section provides an overview of the RAS system to monitor and maintain the Sonexion system.

Reliability, Availability, and Serviceability (RAS) is a set of attributes that reflect the robustness of a hardware and/or software system.

- Reliability is a function of time that expresses the probability at a given future time that a system will still be working, given that it was working at that time.
- Availability is the measure of how often the system is available for use (such as a system's up-time percentage). Availability and reliability may sound like the same concept, but they are different, as a system can have great availability but offer no reliability.
- Serviceability is a broad definition that describes the ease of system service or repair.

Reliability and Availability (RA)

Reliability and availability features keep a system working when failures occur. Examples of RA include:

- Hardware redundancy (power, servers, networks)
- Storage stacks (SCSI, RAID, local file system)
- HA stack
- Clustered file system

Serviceability

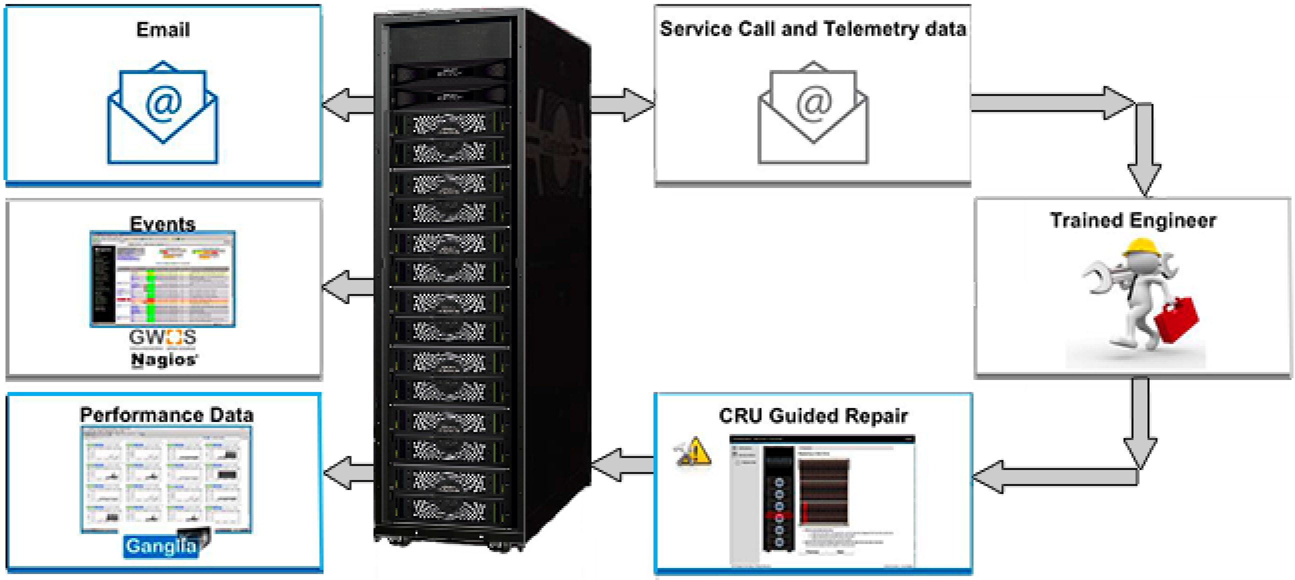
Serviceability consists of on-product capabilities and off-product tools, processes and staff. Serviceability supports the needs of the product (its RA model, for example); HPC and archive products will behave differently. There are cost constraints associated with serviceability. Most important, the customer should not be impacted; there should be no loss of availability for maintenance operations. Serviceability focuses on Level 1 service personnel and eventually, end users.

In Sonexion products, serviceability includes the following capabilities:

- Inventory discovery (part/serial numbers, firmware versions, location)
- Monitoring, diagnostics, fault isolation of:
  - Data collection from functional codes (RAID, HA, FS)
  - Discovery and data collection from hardware (SES, IPMI, SNMP)
- Reporting service manager policy engine (product-specific)
- Reporting service notifications (user email, SNMP)
- Reporting live event telemetry stream (interesting events)

- Guided repair assistance for service
- Concurrent updates - no downtime/interruption to apply RAS-based SUs
- Supports disks, PSUs/PCMs and ESM controller server modules

Figure 7. RAS Service Notification Scenario



RAS for Sonexion provides an "expert" system:

- Architected specifically for Sonexion.
- With topological awareness that enables correct issue prioritization.
- Designed to reduce false positives and repetitive alerts.
- With event-driven data end-to-end across the system, allowing real-time updates.

The Sonexion RAS service console provides a real-time GUI with guided repair assistance.

Remote support includes email notifications, user alert emails, and telemetry data. RAS-based SUs are designed to be applied as concurrent (non-disruptive) updates.

8.1 Configure SMTP Relay, RAS Email, Remote Support, and Service Console

Prerequisites

Before performing any of the procedures in this section, verify that RAS is installed on the system:

```
[MGMT0]$ cscli service_console show
```

The system responds with the current RAS status, either `Service Console enabled` or `Service Console disabled`.

About this task

This section provides procedures to configure SMTP and email notifications, and to enable/disable the service console. Console-related service configuration changes also can be performed via the CSCLI command-line interface.

In this process, the system administrator configures an external SMTP relay so email notifications pertaining to system events and associated system messages can be issued.

Email notifications are sent to users with email addresses configured by the system administrator. This section explains how to configure a list of users to be sent notifications and how to enable/disable an individual user’s notifications.

> **IMPORTANT:** Email notifications to the specified user list are sent only if the notification functionality is enabled.

The service console login can be enabled (configured) to access guided replacement service procedures, such as replacing a disk. Systems are shipped with service console capabilities enabled, by default.

Procedure

Configure SMTP

1. Show the current SMTP configuration:

```
[MGMT0]$ cscli service_console configure smtp show
SMTP relay: mailrelayus.us.company.com:25
```

The example above shows a hypothetical configuration.

2. Configure the SMTP relay:

```
[MGMT0]$ cscli service_console configure smtp relay \
--host=smtp_host --port=smtp_port
```

where: *`smtp_host`* (required field) is the SMTP relay host for email notifications, and *`smtp_port`* is the SMTP relay port (default: 25).

Use the `--help` flag to obtain additional information for configuring the SMTP relay. For example:

```
 [MGMT0]$ cscli service_console configure smtp relay --help
```

Configure the Email Notification List

This section explains how to configure a list of users to be sent notifications and how to enable/disable individual user notifications.

> **IMPORTANT:** Email notifications to the specified user list are sent only if the notification functionality is enabled.

3. Display a list of users that are already in the email notification list:

```
[MGMT0]$ cscli service_console configure email show
```

4. Add an email address:

```
[MGMT0]$ cscli service_console configure email add –A user@domain.com
```

Delete an existing email address:

```
[MGMT0]$ cscli service_console configure email delete –A user@domain.com
```

IMPORTANT: Email notifications to the specified user list are sent only if the notification functionality is enabled.

5. Verify that the email address has been correctly established:

```
[MGMT0]$ cscli service_console configure email show
```

6. Enable user email support:

```
[MGMT0]$ cscli service_console configure email enable
```

(To disable user email support):

```
[MGMT0]$ cscli service_console configure email disable
```

7. Send a test email to verify that the email address has been properly added:

```
[MGMT0]$ cscli service_console configure email send_test_email
```

If the SMTP relay is configured properly, the configuration emails are enabled, and the specified email addresses have been set up to receive notifications, the following test email message is sent to the user list:

```
Subject: Remote Service TEST: cls12345
Body:
System Identification:
       System Serial Number: cls12345
       System Identifier:
Current System Time: Tue, 21 Oct 2017 18:05:54
System Timezone: PDT
TEST: This email is to test system SMTP relay setup only, please
ignore.
```

Example of an email successfully sent to remote support:

```
Successfully sent test email to remote support with subject
'Remote Service TEST: cls12345'
```

Example of an email successfully sent to users whose email addresses are properly configured:

```
Successfully sent test email to configured user(s) with subject
'Remote Service TEST: cls12345'
```

Enable or Disable the Service Console

8. Verify that the service console is enabled:

```
[MGMT0]$ cscli service_console enable
```

IMPORTANT: Sonexion are shipped with service console capabilities enabled, by default.

(To disable the service console:)

```
[MGMT0]$ cscli service_console disable
```

8.2 Configure Service Console Access

Prerequisites

Software version 3.0.0 or higher is installed.

About this task

An important tool in Sonexion RAS solution is its GUI-based service console interface. It shows system issues and conditions detected by the RAS solution, which can be reported as email notifications to users and Cray support, and lists system repair actions to be completed by service personnel.

By default, Sonexion systems ship with service console login capabilities enabled. To maintain a more secure environment, system administrators may opt to restrict access to the service console, and enable logins to the interface only when a service operation needs to be performed.

Disabling the service console only restricts logins to the GUI; no other aspect of the Sonexion RAS system is affected. A change (enable/disable action) to the service console login takes approximately 30 seconds to be completed.

Procedure

1. Enable login to the service console:

   ```
   [MGMT0]$ cscli service_console enable
   ```

2. To disable login to the service console:

   ```
   [MGMT0]$ cscli service_console disable
   ```

3. Show the status of the service console (enabled or disabled):

   ```
   [MGMT0]$ cscli service_console show
   ```

8.3 Configure RAS System Settings

Prerequisites

RAS is installed.

About this task

This section specifies the Sonexion installation name (identifier) for the RAS system. Use the `system` command to configure this value or show the currently specified identifier.

Procedure

1. Show the current system identifier:

```
[MGMT]$ cscli service_console configure system show
```

2. Set the system identifier:

```
[MGMT]$ cscli service_console configure system identifier name
```

   where *name* (a required field) is the name of the Sonexion system.

8.4 Configure the REST API

Prerequisites

RAS is installed in the system.

About this task

This section describes how to enable or disable RAS data streaming via the REST API, with the option to show current status.

Procedure

1. Enable REST API access:

```
[MGMT]$ cscli service_console configure rest_api enable
```

2. Disable REST API access:

```
[MGMT]$ cscli service_console configure rest_api disable
```

3. Show the status of data streaming via the REST API (enabled or disabled):

```
[MGMT]$ cscli service_console configure rest_api show
```

8.5 Interpret RAS Email Notifications

This section describes the types of RAS email notifications sent to users. The service console can be configured to send email notifications to specific users about service event messages (SEMs), which report events related to the service console’s operation and include detail and location information about failed components. SEM notifications are sent to a pre-configured user email list specified by the system administrator.

SEMs include details and location information of degraded or failed components requiring repair. The notification includes a human-readable message, with instructions for accessing the service console, and a JSON-formatted machine-readable attachment.

SEM contents include:

- System serial number and identification information
- Date, timestamp and time zone of the service event
- Human-readable description of the failed component
-
    - Serial number, model, manufacturer, and firmware level (if applicable)
    - Location information (rack and enclosure containing the faulty part)
- Machine-readable attachment

Files attached to SEMs are used to provide automated FRU replacement part number lookups.

The following sample SEM (message body and JSON-formatted attachment) lists service events for a failed disk and power supply unit, along with the service console's IP address for service personnel to log in and repair the failed hardware.

SEM Body

Subject:

```
Remote Service SEM: SNX0G1GXC
```

Body:

```
This is a Sonexion service event notification:

The service console can be accessed from the service port via this URL: http://
172.16.2.1:8080

System Identification:
     System Serial Number: SNX0G1GXC
     System Identifier:

Current System Time: Thu, 27 Nov 2014 18:46:54 System Timezone: EST

Current outstanding service call events:

Item 1 : "Disk drive needs replacement"

Service Code: 002005001
Time the event was first detected: Thu, 27 Nov 2014 09:31:28 EST

Details of failed component

     Disk Serial #: S0M122HN0000B40298QD
     Disk Model : ST600MM0006
     Drive Manufacturer: SEAGATE
     Firmware version of drive at time of failure: XLGD
     T10 Enabled: "Yes"

Location of failed component

     Rack Name: Rack1
     Enclosure Model: 2U24
     Enclosure Location: 36U
     Disk located in slot: 15

Item 2 : "Power supply issue detected"
```

```
Service Code: 002005003
Time the event was first detected: Thu, 27 Nov 2014 09:49:27 EST

Details of failed component

     Power supply chassis type: OneStor Power One 580W_AC_PCM, P/N 82562.
     Power supply part number: 0082562-07
     Power supply product version: 0311
     Power supply serial number: PMW8256200052C7

Location of failed component

     Rack Name: Rack1
     Enclosure Model: 2U24
     Enclosure Location: 36U
     Power supply located in bay: 0
```

SEM Attachment

Attachment:

```
sem-SNX0G1GXC-2014-11-27_18-46-54_074697.json
```

```
{
    "system_serial_number":
    "SNX0G1GXC",
    "system_identifier": "",
    "system_time": "Thu, 27 Nov 2014 18:46:54",
    "utc_timestamp": 1417132014,
    "system_timezone": "EST", "local_time": "Thu, 27 Nov 2014 18:46:54 EST",
    "service_console_url": "http://172.16.2.1:8080",
    "version": 4,
    "messages": [
    {
        "event_description": "Disk drive needs replacement",
        "re_event_code": "001001001",
        "uuid": "54ac138e-7642-11e4-8045-000c293bad8b",
        "confirmed_time": 1417098798.773304,
        "creation_time": 1417098798.844036,
        "dcs_timestamp": "1417098688",
        "completion_time": 0.0,
        "state": "SVC_CREATED",
        "location": {
              "index": "15",
              "enclosure_location": "36U",
              "enclosure_model": "2U24",
              "rack": "Rack1"
        },
        "event_code": "002005001",
        "disk": {
            "dm_report_t10": "11110111100",
            "firmware": "XLGD",
            "t10_enabled": true,
            "serial_number": "S0M122HN0000B40298QD",
            "model": "ST600MM0006",
            "manufacturer": "SEAGATE"
        },
        "type": "disk"
},
{
        "event_description": "Power supply issue detected",
        "re_event_code": "001003001",
        "uuid": "93c895bc-7677-11e4-ab16-000c293bad8b",
        "confirmed_time": 1417121643.868457,
        "creation_time": 1417121668.054007,
        "dcs_timestamp": "1417099767", "completion_time": 0.0,
        "state": "SVC_CREATED",
        "location": {
           "index": "0",
           "enclosure_location": "36U",
           "enclosure_model": "2U24",
           "rack": "Rack1"
        },
        "event_code": "002005003",
        "type": "power_supply",
        "power_supply": {
           "status": "failure_detected",
           "serial_number": "PMW8256200052C7",
           "part_number": "0082562-07",
           "product_version": "0311",
           "chassis_type": "OneStor Power One 580W_AC_PCM, P/N 82562."
        }
    }
    ]
}
```

8.6 Use Guided Replacement Procedures via the Service Console

Prerequisites

RAS is installed and enabled.

About this task

This section provides steps to access the service console and follow the guided replacement procedures.

Procedure

Connect to the Service Console

1. Cable a laptop to any available port on either management switch.
2. Navigate to http://service:8080 using a web browser:

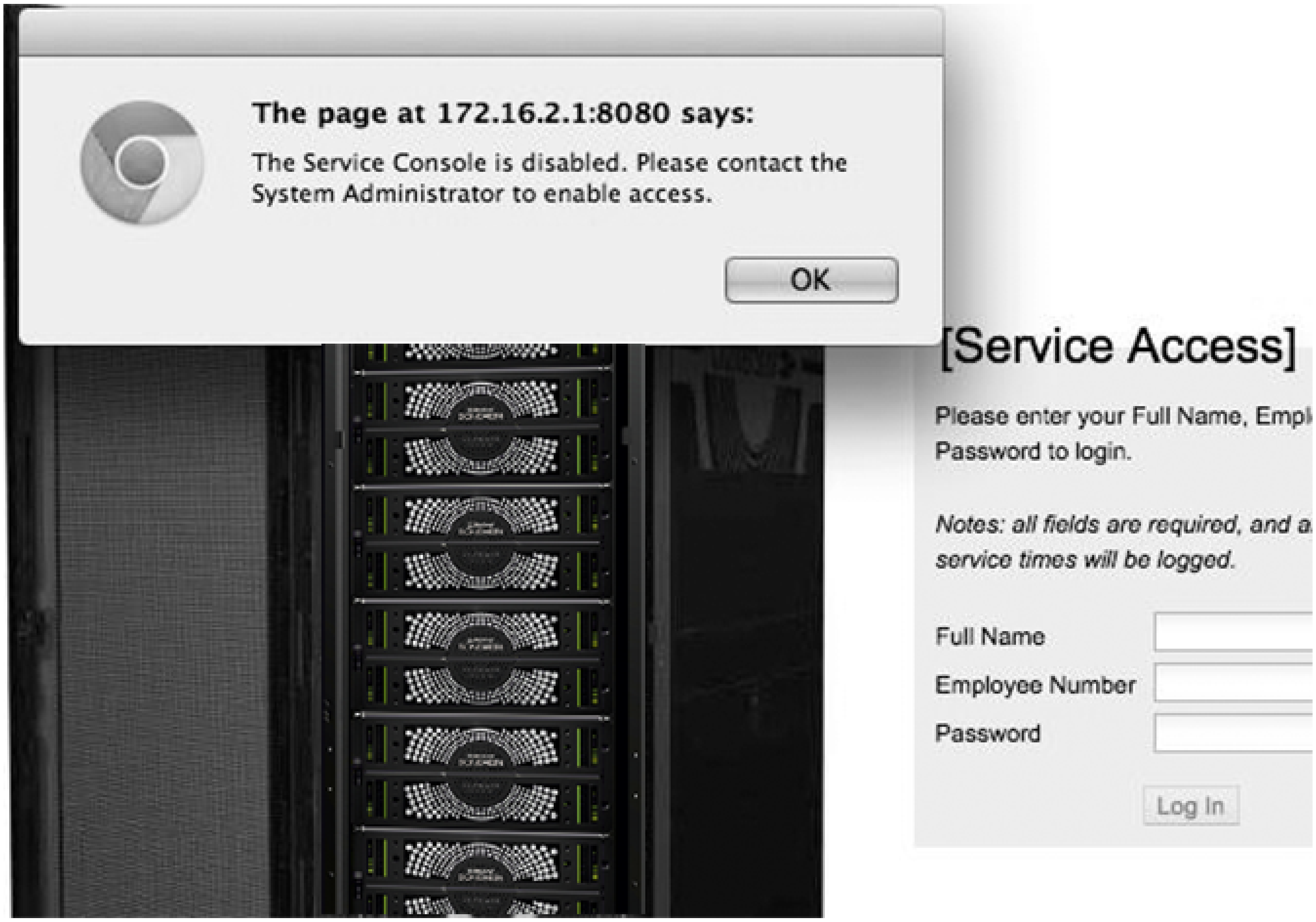
   If that URL is not active, log into port 8080 of the IP address of the currently active MGMT node (MGMT0).

   ```
   http://IP address of the currently active (primary) MGMT node:8080
   ```

3. Enter authentication credentials provided by the customer's service center.

IMPORTANT: If, when attempting to log into the service console, a message similar to the dialog box shown appears, then the system administrator has disabled access to the interface and service actions can not proceed. If the system administrator re-enables the service console access, the **Log In** button becomes active within 30 seconds and logging in can continue. For more information, refer to the section, *Configure Service Console Access*.

Figure 8. Dialog Box Indicates Disabled Service Console



The page at 172.16.2.1:8080 says:

The Service Console is disabled. Please contact the System Administrator to enable access.

[Service Access]

Please enter your Full Name, Empl
Password to login.

Notes: all fields are required, and a
service times will be logged.

Full Name
Employee Number
Password

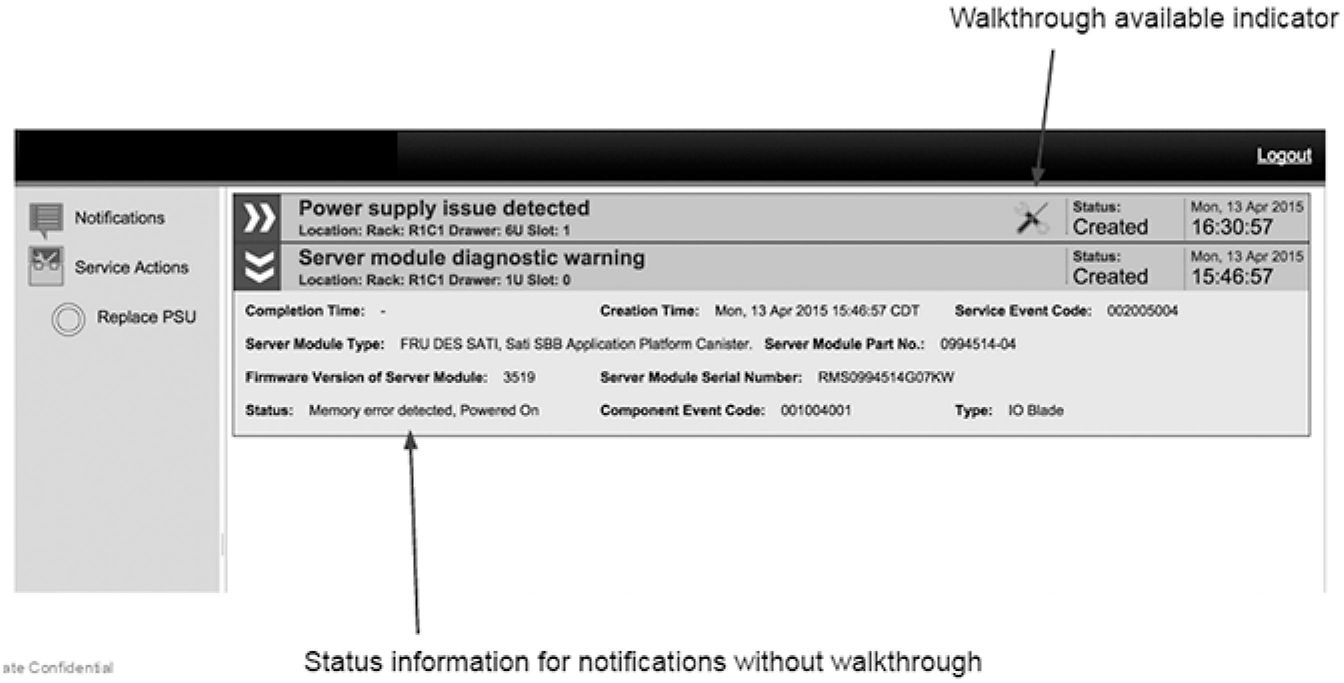
When the service console is logged into, it displays a list of repair actions (usually failed hardware replacements). The service console provides a step-by-step guide through the repairs to successfully return the system to normal operating mode.

Guided Replacement Procedures

4. Read the list of repair actions.

Diagnostic notifications and service actions are displayed. Serviceable items that can be addressed using a guided replacement procedure (a “walkthrough”), are identified with an "available walkthrough" indicator and appear in a pick list located on the left side of the service console, as shown.

Figure 9. Service Console: Notification View



Walkthrough available indicator

Power supply issue detected
Location: Rack: R1C1 Drawer: 6U Slot: 1
Status: Created
Mon, 13 Apr 2015 16:30:57

Server module diagnostic warning
Location: Rack: R1C1 Drawer: 1U Slot: 0
Status: Created
Mon, 13 Apr 2015 15:46:57

Completion Time: - **Creation Time:** Mon, 13 Apr 2015 15:46:57 CDT **Service Event Code:** 002005004
Server Module Type: FRU DES SATI, Sati SBB Application Platform Canister. **Server Module Part No.:** 0994514-04
Firmware Version of Server Module: 3519 **Server Module Serial Number:** RMS0994514G07KW
Status: Memory error detected, Powered On **Component Event Code:** 001004001 **Type:** IO Blade

Status information for notifications without walkthrough

5. Click on a specific notification to expand the view and access additional information about the service event.

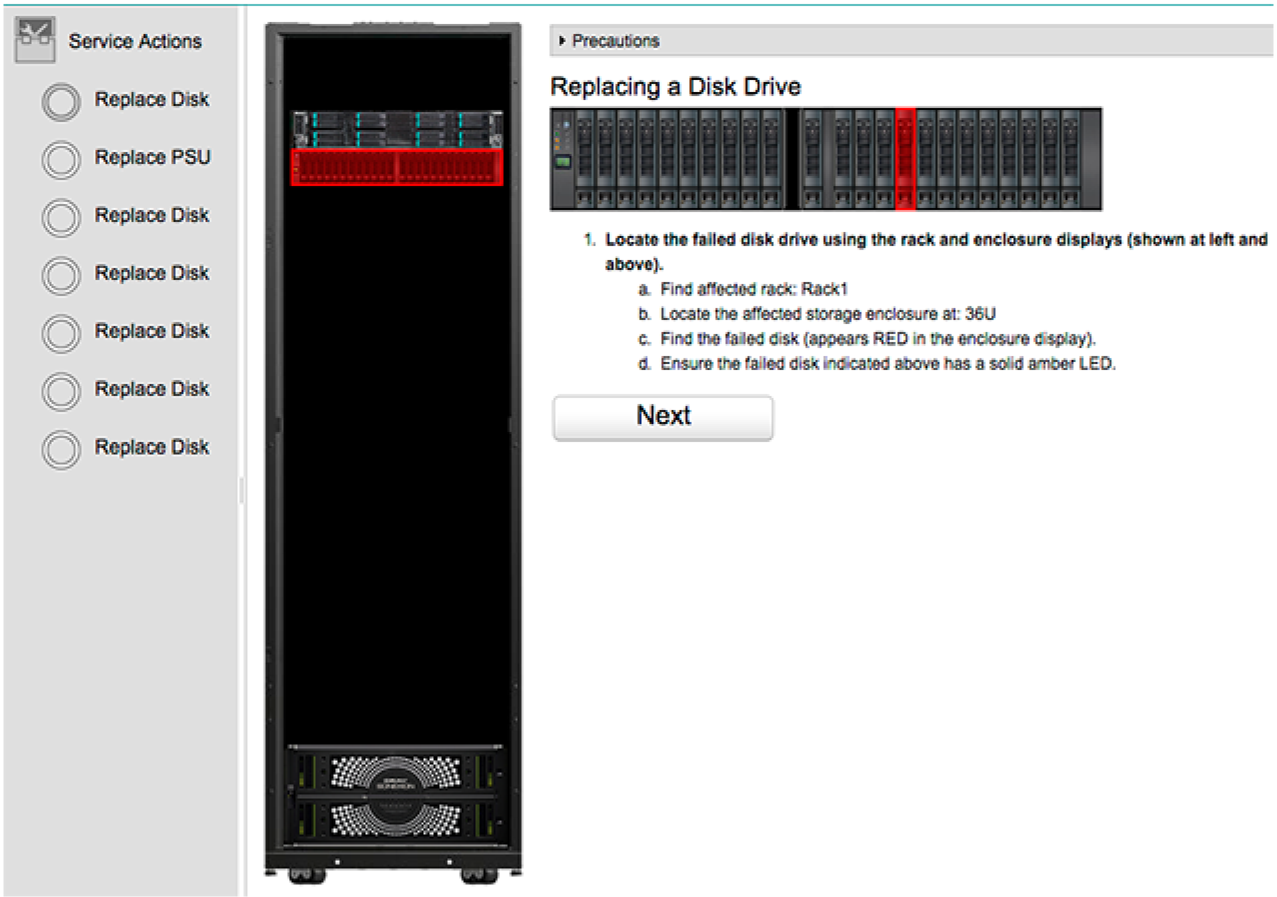
Selecting a service action from the pick list launches the guided replacement procedure for the specified failed hardware component. Walkthrough steps are displayed to the right of the pick list and feature:

- Visualization of the system
- Clear identification of the affected chassis
- Clear identification of the affected FRU
- Clear and concise instructions on how to perform the replacement procedure
- Live feedback during a procedure

Each service action provides visual cues (in red) to help the technician identify the affected enclosure and the specific location of the degraded or failed component. The following figures illustrate the guided replacement procedure to replace a failed DDIC in a 2U24 enclosure, with red indicators showing the affected enclosure in the rack and the location of the failed disk in the enclosure.

6. If the guided replacement includes precautions or warnings related to the repair, the technician is prompted to acknowledge them before starting the procedure.

Figure 10. Service Console: Guided Replacement Procedure for a Disk Drive in a 2U24 Enclosure

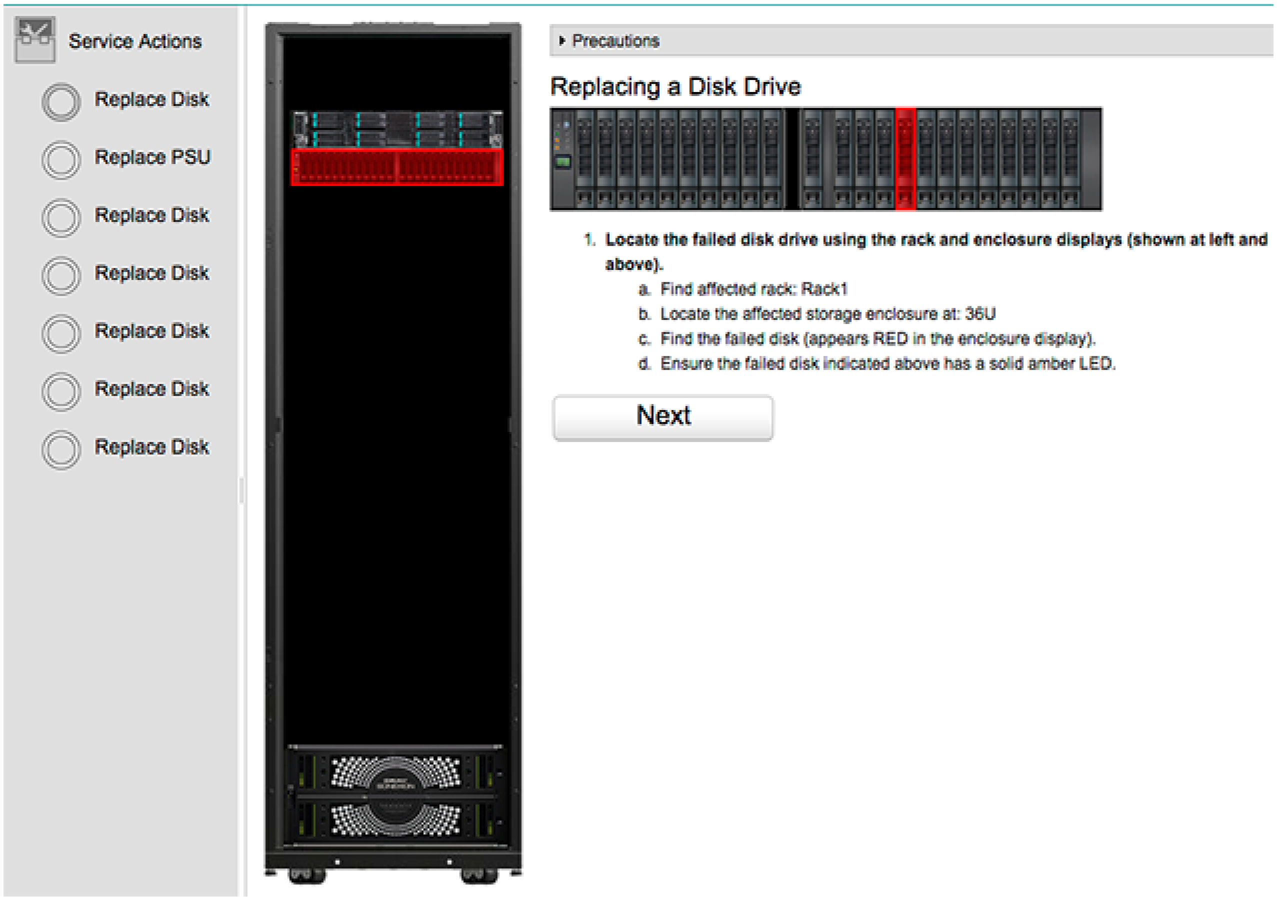


Replacing a Disk Drive

1. **Locate the failed disk drive using the rack and enclosure displays (shown at left and above).**
   a. Find affected rack: Rack1
   b. Locate the affected storage enclosure at: 36U
   c. Find the failed disk (appears RED in the enclosure display).
   d. Ensure the failed disk indicated above has a solid amber LED.

7. The system validates the new hardware.

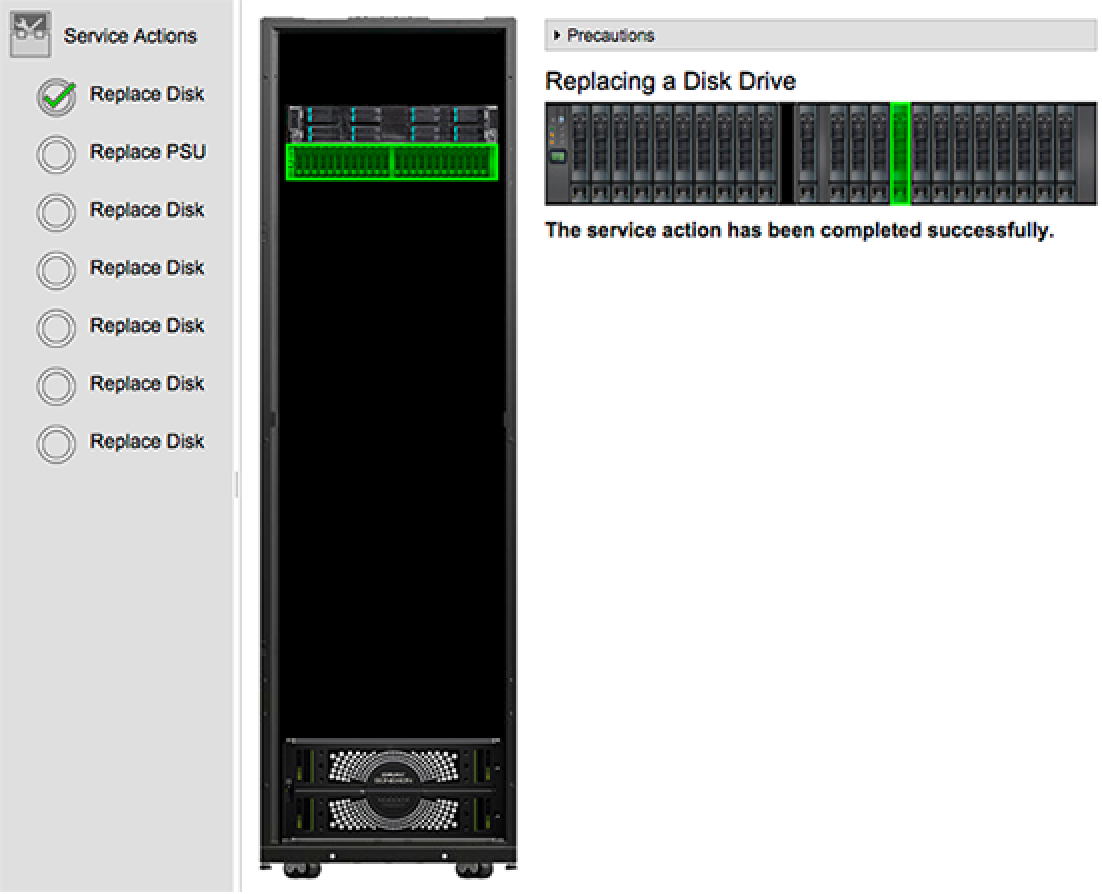
Figure 11. Service Console: Guided Replacement Procedure to Replace a Disk Drive



Replacing a Disk Drive

1. **Locate the failed disk drive using the rack and enclosure displays (shown at left and above).**
   a. Find affected rack: Rack1
   b. Locate the affected storage enclosure at: 36U
   c. Find the failed disk (appears RED in the enclosure display).
   d. Ensure the failed disk indicated above has a solid amber LED.

Figure 12. Service Console: Successful Completion of a Disk Drive Replacement



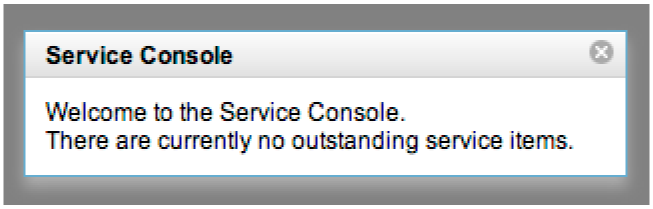
Replacing a Disk Drive

The service action has been completed successfully.

If the repair was completed correctly, the service action's icon updates to green, in addition to visual cues appearing within the FRU replacement procedure. The following screenshot illustrates an example of a complete, successful disk drive replacement using the guided procedure on the service console.

8. When all service actions are complete, the following message appears:

   Figure 13. Service Console: Status Notification



Service Console

Welcome to the Service Console.
There are currently no outstanding service items.

9. Log out of the service console.

If required, notify the system administrator that all repairs have been made and the service console login can be disabled.

8.7 service_console Commands

Modes: Daily

The service_console command lets a site administrator manage the system service console and block or allow its use for a service technician.

The service console is an alternate system management interface that is limited to maintenance functions, and is for use by authorized service personnel only. Even though it bypasses the normal administrative login, the service console cannot be used to access the root account, or to access data on the Lustre file system.

While it is not necessary to disable the service console for any system that is physically secure, administrators in high-security environments might prefer to disable it. In that case, they will need to re-enable the service console each and every time a service technician arrives on site to perform system maintenance, including replacement of failed disk drives and other parts.

Configuration consists of two main areas

- Configuring the SMTP relay, this allows the system to send emails to users and service emails.
- Configuring user email addresses to allow service email to go to system users.

This following commands and available subcommands are used to manage the service console.

Synopsis

```
$ cscli service_console [-h] {enable,disable,show,notifications,configure}
```

where:

Subcommands	Description
enable	Enables the service console
disable	Disables the service console
show	Show service console enabled status
notifications	View RAS notifications
configure	Configure RAS notifications

Optional Arguments	Description
-h |--help	Displays the help message and exits

Examples:

Enable the service console.

```
$ cscli service_console enable
Service Console enabled.
```

Disable the Service console.

```
$ cscli service_console disable
Service Console disabled.
```

Show the status of the service console (enabled).

```
$ cscli service_console show
Service console enabled.
```

Show service console notifications (service events).

```
$ cscli service_console notifications show
Current outstanding service call events:

Item 1 : "Disk drive needs replacement"

Service Code: 002005001
Time the event was first detected: Wed, 17 Jun 2015 17:18:14 EDT

    Details of failed component
    Disk Serial #: S0M122HN0000B40298QD
    Disk Model: ST600MM0006 Drive
    Manufacturer: SEAGATE
    Firmware version of drive at time of failure: XLGE T10     Enabled: Yes

Location of failed component

    Rack Name: Rack1
    Enclosure Model: 2U24
    Enclosure Location: 36U
    Disk located in slot: 15

Item 2: "Power supply issue detected"

Service Code: 002005003
Time the event was first detected: Wed, 02 Dec 2015 12:58:57 PST

Details of failed component

    Power supply chassis type: OneStor Power One 764W_AC_PCM (Original Silver DFM).
    Power supply part number: 0945768-10
    Power supply product version: 0314
    Power supply serial number: PMW0945768J1BDY

Location of failed component

    Rack Name: R1C1
    Enclosure Model: 2U24
    Enclosure Location: 24U
    Power supply located in bay: 1
```

Show service console configuration options.

```
$ cscli service_console configure
Commands:
    smtp            Configure SMTP settings
    snmp            Configure SNMP settings
    email           Configure email for sending RAS notifications
    system          Configure system settings
    rest_api        Configure REST API settings
```

`notifications` Subcommand

The `notifications` command is a subcommand of the `service_console` command, and is used to view notifications.

Synopsis

```
$ cscli service_console notifications -h {show}
```

Subcommands	Description
show	Show outstanding notifications.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

Service Console Configuration Commands

These commands are used to configure the SMTP relay, SNMP, remote support, email addresses for RAS notifications, system and REST API settings.

Synopsis

```
$ cscli service_console configure {smtp,snmp,remote_support,email,system,rest_api}
```

Subcommands	Description
smtp	Configures SMTP settings
snmp	Configures SNMP settings
remote_support	Configure remote support RAS notifications
email	Configure email for sending RAS notifications
system	Configure system settings
rest_api	Configure REST API settings

Optional Arguments	Description
-h |--help	Displays the help message and exits

`smtp` Commands

The `smtp` command is a second-level subcommand of the `service_console` command, and is used to configure SMTP.

Synopsis

```
$ cscli service_console configure smtp -h -H {relay,show}
$
```

Subcommands	Description
relay	Configures SMTP relay for sending RAS notifications
show	Displays current SMTP configuration

Optional Arguments	Description
`-H |--host`	SMTP host (required)
`-h |--help`	Displays the help message and exits

`relay` Command

The `relay` command is the third-level subcommand of the `service_console` command, and is used to configure SMTP relay.

Synopsis

```
$ cscli service_console configure smtp relay [-h] -H smtp_host -P smtp_port
```

Subcommands	Description
`-P` *`smtp_port`* `|--port` *`smtp_port`*	SMTP port [default: 25]
`-H` *`smtp_host`* `|--host` *`smtp_host`*	SMTP host (required)
`-h |--help`	Displays the help message and exits

Examples:

Configure the SMTP relay.

```
$ cscli service_console configure smtp relay -H
mailrelayus.xyus.xyratex.com
OK.
```

Show the current SMTP configuration.

```
$ $ cscli service_console configure smtp show
SMTP relay: mailrelayus.xyus.xyratex.com:25
```

`snmp` Command

The `snmp` command is a second-level subcommand of the `service_console` command, and is used to configure SNMP settings

Synopsis

```
$ cscli service_console configure snmp -h {enable,disable,show}
```

Subcommands	Description
`enable`	Enables SNMP monitoring
`disable`	Disables SNMP monitoring
`show`	Shows current SNMP settings

Optional Arguments	Description
`-h |--help`	Displays the help message and exits

`remote_support` Commands

The `remote_support` command is a second-level subcommand of the `service_console` command, and is used to configure SMTP remote support.

Synopsis

```
$ cscli service_console configure remote_support -h {enable,disable,show}
```

Subcommands	Description
`enable`	Enables remote support
`disable`	Disables remote support
`show`	Shows current remote support status

Optional Arguments	Description
`-h |--help`	Displays the help message and exits

Examples:

Disable remote support.

```
$ cscli service_console configure remote_support disable

Remote Support disabled.
```

Display remote support status.

```
$ cscli service_console configure remote_support show
Remote Support enabled.
```

Enable remote support.

```
$ cscli service_console configure remote_support enable
Remote Support enabled.
```

Email Notification Commands

This command configures email notifications, including enabling or disabling notification functionality, adds or deletes email addresses to the user list, shows user list entries and sends test email to determine if user notifications can be sent successfully.

Synopsis

```
$ cscli service_console configure email [-h]
{show,add,delete,enable,diable,send_test_email}
```

Subcommands	**Description**
`show`	Shows all notification email addresses
`add`	Adds a notification email address. Also specify the email address to be added (required): `-A` *`email_address`* `|--address` *`email_address`*
`delete`	Deletes a notification email address Also specify the email address to be added (required): `-A` *`email_address`* `|--address` *`email_address`*
`enable`	Enables email notifications
`disable`	Disables email notifications
`send_test_email`	Sends test email to all enabled services

Optional Arguments	**Description**
`-h |--help`	Displays the help message and exits

Examples:

Show email addresses configured to receive RAS notifications (no configured addresses).

```
$  cscli service_console configure email show
No email addresses are configured to receive service notifications.
```

Show email addresses configured to receive RAS notifications (several configured addresses).

```
$  $ cscli service_console configure email show
The following email addresses are configured to receive service notifications:
sam_jones@xyzcorp.com
amy_cooper@xyzcorp.com
```

Add an email address to the user list (success).

```
$ cscli service_console configure email add –A sam_jones@xyzcorp.com
Successfully added 'sam_jones@xyzcorp.com' to receive notification emails.
```

Add an email address to the user list (fails because it is currently associated with the user list).

```
$ cscli service_console configure email add –A sam_jones@xyzcorp.com
Unable to add: 'sam_jones@xyzcorp.com' is already configured for receiving
notification emails.
```

Delete an email address from the user list (success).

```
$ cscli service_console configure email delete –A sam_jones@xyzcorp.com
Successfully removed 'sam_jones@xyzcorp.com' from receiving notification emails.
```

Delete an email address from the user list (fails because it is not associated with the user list).

```
$ cscli service_console configure email delete –A sam_jones@xyzcorp.com
Unable to delete: 'sam_jones@xyzcorp.com' is not configured for receiving
notification emails.
```

Enable email notifications.

```
$ cscli service_console configure email enable
OK.
```

Disable email notifications.

```
$ cscli service_console configure email disable
OK.
```

system Command

The system command is a second-level subcommand of the service_console command, and is used to configure system settings.

Synopsis

```
$ cscli service_console configure system [-h] {identifier,show}
```

Subcommands	Description
identifier	Sets the system identifier
show	Shows the current system identifier (name)

Optional Arguments	Description
-h |--help	Displays the help message and exits

Examples:

Show the current system identifier (name specified).

```
$ cscli service_console configure system show
System settings:
    System identifier name: snx-002
```

Show the current system identifier (no name set).

```
$ cscli service_console configure system show
```

identifier Command

The identifier command is a third-level subcommand of the service_console command, and is used to set the system identifier name.

Synopsis

```
$ cscli service_console configure system identifier -h -n id_name
```

Usage:

```
$ cscli service_console [options] configure [options]
system [options] identifier [options]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits
`-n` *`id_name`* `|--name=`*`id_name`*	Name (required)

Example:

Set the system identifier.

```
$ cscli service_console configure system identifier -name=snx-002
System identifier name has been set to: snx-002
```

`rest_api` Command

The `rest_api` command is a second-level subcommand of the `service_console` command, and is used to configure REST API settings.

Synopsis

```
$ cscli service_console configure rest_api -h {enable,disable,show}
```

Subcommands	Description
`enable`	Enables the REST API
`disable`	Disables the REST API
`show`	Shows current status of the REST API

Optional Arguments	Description
`-h |--help`	Displays the help message and exits

9 Sonexion System Snapshot Analyzer

9.1 SSA Introduction

The System Snapshot Analyzer (SSA) software supports analytics technology for Sonexion® systems.

SSA collects, analyzes, and uploads (if uploading is enabled) product health, configuration, or triage information to Cray. Uploaded information captured by SSA is analyzed using a sophisticated analytics platform that detects and enumerates changes over time, changes in the health state of the system, and processes triage information to assist with case resolution. Through automation, SSA is designed to improve the overall customer experience by reducing the manual effort and time required to report and resolve support issues.

The SSA Sonexion shepherd client software manages the collection, first-level analysis, and secure transport of support information back to Cray.

For additional information and references on SSA, please visit *http://www.cray.com/support*. For details on how to activate an SSA account and download SSA software, refer to Cray SFDC/Case Management Knowledge *Article 4546*, *Getting Started with the Cray System Snapshot Analyzer (SSA)*.

Request Support for SSA

To request support, contact a Cray support representative or file a service case against the Sonexion SSA component. Please submit request for enhancements (RFEs) or defect reports (bugs) against SSA. Feedback and suggestions are valued and welcomed.

9.2 Configure SSA for Sonexion

Download the Sonexion SSA 3.0.0 shepherd software by following guidance in *Cray Knowledge Article 4546*, *Getting Started with the Cray System Snapshot Analyzer (SSA)*.

A CrayPort account is required to download software and activate an SSA account. SSA account activation is required to obtain authentication credentials for use during configuration and to enable snapshot uploads to Cray.

Visit *https://crayport.cray.com* if you have active support entitlements and would like to register for CrayPort accounts.

Shepherd Overview

Sonexion SSA shepherd software is delivered in two packages. One contains the shepherd command line interface (`ssacli`) and libraries. The other contains plugins for a specific Cray platform (e.g., Cray XE, Cray XC, or Sonexion). These packages are revision matched, with the plugin package dependent on the base shepherd software.

The shepherd operates in three, progressive stages—collect, snapshot, and upload.

Collect Stage The collect stage is responsible for utilizing plugins to perform collection and analysis of system information, and to prepare for subsequent stages.

Snapshot Stage The snapshot stage searches for collections which have not been previously processed. It encodes them into a network friendly format in preparation for upload to Cray.

Upload Stage The upload stage then takes any snapshots which have not been uploaded and attempts to upload them to Cray for further analysis and processing.

The upload takes place over a secure network connection, using transport layer security (TLS), and is further authenticated using credentials from customer SSA accounts.

The shepherd uses a configurable purge policy to remove older collections or snapshots.

The shepherd is either invoked on-demand or automatically on a schedule via `cron`. Information on supported run configurations of the shepherd and their purpose are covered in *Collect and Upload a Snapshot* on page 56.

9.2.1 SSA Prerequisites

SSA Upload Account Activation

The activation of an SSA account in CrayPort is required in order to obtain the organization name and passphrase to use in the configuration of SSA for the upload of snapshot information to Cray.

Network Connectivity

The shepherd application only initiates outbound network connection—over TCP/IP (version 4) TCP port 443—to the network host `ssa.cray.com`. Cray does not initiate an inbound connection to the customer network/system. The outbound network session is established only long enough to submit a snapshot of information to Cray. Then it terminates.

To communicate with the Cray upload system, outbound connectivity must be provided from the user site as described. Optionally, a local network proxy (HTTP, SOCKS) can be utilized through features in the shepherd application. For additional details about how to configure a proxy, refer to the comments in the `shepherd.conf` file distributed with the shepherd.

Configuration of the `sudo` Utility

The shepherd leverages the functionality provided by the `sudo` utility to remove privileges and provide an audit trail of application activity. The `root` user must be able to execute any command, as any user, on installation nodes via `sudo`. Sudo must be configured so that the `root` user is not prompted for a password to execute `sudo`.

Disable `requiretty`

In addition, `requiretty` must be disabled to enable `sudo` execute scheduled jobs via SSA (e.g., those not run directly on a real TTY, by an interactive user).

9.2.2 Install the Sonexion SSA RPM Packages

Prerequisites

This procedure must be performed as the `root` user on the cluster management node.

About this task

This procedure installs the base SSA shepherd application and system-specific plugins for systems.

Time Required: Approximately 30 minutes. SSA RPMs can be installed during customer operations and while the Lustre file systems are running.

Packages for Sonexion are used in the examples for illustration only. Always download the latest shepherd software specific to the system.

Perform the RPM installation steps on the primary management node, then on the secondary management node. The SSA software is not synchronized between the two management nodes. The configuration and SSA client (shepherd) versions must be the same on each management node.

The SSA installation process creates cron entries in `/etc/cron.d/cray-ssa/` for scheduled operation.

Procedure

1. Log in to the primary management node as `root`.

2. If upgrading SSA, disable SSA using *Enable or Disable SSA*.

3. Install the RPM packages:

   ```
   # rpm -ivh \
   cray-ssa-shepherd-1.7.0-0.x86_64.rpm cray-ssa-shepherd-sonexion-plugins-1.7.0-0.x86_64.rpm
   Preparing...                          ################################# [100%]
      1:cray-ssa-shepherd        ################################# [ 50%]
   Active mgmt node, attempting to create /mnt/mgmt/var/opt/cray/ssa directory structure ...
   Setting Alternatives (update-alternatives) ...
      2:cray-ssa-shepherd-sonex################################# [100%]
   ```

   Installation of the RPMs yields messages associated with the `alternatives` software. For information about how to manage the version of shepherd used, refer to the `man` page for `update-alternatives` and *Specify a Different Version of SSA* (if the RPM for the platform uses `alternatives`).

4. Repeat the steps above on the secondary management node.

5. Exit and `sudo` to `root` on each management node to invoke the new `root` user environment.

9.2.2.1 Edit the Sonexion Shepherd Configuration File

Prerequisites

This procedure must be performed as the `root` user.

About this task

The Sonexion shepherd uses a single configuration file, `/opt/cray/ssa/default/etc/shepherd.conf`. The configuration file is structured into sections and contained within square brackets, for example `[control]`. Each section and each of its related parameters contain a header of descriptive configuration information. Sections may contain individual or groups of related parameters. Prior to running the shepherd, edit this file on the primary management node.

The steps below describe commonly configured parameters for desired shepherd features. It is imperative that both the primary and secondary management nodes have the same shepherd configuration entries. After the completion of the procedure below on the primary management node, copy the configuration file over to the secondary management node.

Always make a backup copy of the shepherd configuration file `(shepherd.conf`) for reference.

> **IMPORTANT:** Review the SSA release notes for versions that have compatible configuration files. If the version being installed is compatible with the previous version, simply make a backup copy of the new version's configuration file and overwrite it with the existing shepherd configuration file.

Procedure

1. Login to the primary management node as `root`.

   - If using an existing compatible configuration file, copy the file to the appropriate location as shown and skip to step 8.

     ```
     MGMT0#  cd /opt/cray/ssa/default/etc
     MGMT0#  cp -a shepherd.conf shepherd.conf-dist
     MGMT0#  cp -a /opt/cray/ssa/ssa_version/etc/shepherd.conf .
     ```

   - If this is an initial installation, proceed to step 2.

2. From the primary management node, edit `/opt/cray/ssa/default/etc/shepherd.conf` file.

3. Enable the shepherd master operation mode. This configuration setting allows all stages (collection, snapshot, upload) to be active.

   Set `master_enabled: true` in the `[control]` section.

   ```
   [control]
   master_enabled: true
   ```

4. Set system identification information in the `[sysinfo]` section.

   a. Set the serial number of the system.

   ```
   [sysinfo]
   serial_num: 99999
   ```

   b. Set the system type.

   ```
   [sysinfo]
   system_type: SNX3000
   ```

   c. Set the system name.

```
[sysinfo]
system_name: prod
```

d. Set a short system description.

```
[sysinfo]
system_description: SNX3000 PROD
```

5. Set upload information in the `[upload]` section.

> **IMPORTANT:** If SSA must be run in local mode, (no information is uploaded to Cray), see *Configure SSA for Local Only Mode* and skip steps 5 through 7.

The `upload_server`, `upload_org`, and `upload_pw` parameters in the `[upload]` section must be set before using the snapshot or upload stages.

a. Set the organization received when the SSA account was activated that snapshots will be uploaded for.

```
[upload]
upload_org: cto01
```

b. Set the password received when the SSA account was activated.

```
[upload]
upload_pw: SuperSecretPassword
```

These parameters must be set for a successful authentication and upload of SSA information to Cray.

6. Optional: Set, in the `[upload]` section, the upload server (`upload_server`) address to an IP address. Or, if using IP name resolution, set an additional X.509 (SSL/TLS) certificate validation option.

a. Optional: If using DNS or a local resolution method (e.g., `/etc/hosts`), set `verify_x509_host` to `true`.

This enables additional protection within the shepherd to validate that the subject name in the X.509 SSL/TLS certificate matches that of the server (ssa.cray.com), allows resolution of the DNS name ssa.cray.com, adds a small amount of security to the upload process, and should be enabled if using a suitable resolution method. The certificate chain for the SSA upload system is maintained locally within the shepherd application (isolated from other certificate stores on the system). The CA bundle file is located in `/opt/cray/ssa/default/etc/ssl/ssa.pem`.

```
[upload]
verify_x509_host: true
```

b. Optional: If not using DNS and if a manually configured local resolution method is not desired, set the `upload_server` to an IP address.

The upload system uses, at the time of this writing, a single IP address, `136.162.62.191`. This IP address should resolve via a DNS `PTR` reference to an `A` record ending in `.cray.com`. Any changes in SSA upload addressing will be communicated directly to customers by Cray.

1. Set `upload_server` to IP address (using the documented IP above).

```
[upload]
upload_server: 136.162.62.191
```

2. Ensure that `verify_x509_host` is set to false.

```
[upload]
verify_x509_host: false
```

7. Save the file and exit.

After the configuration file is saved (with `master_enabled: true`) the `cron` schedule for the shepherd will be activated. The `cron` schedule can be located in `/opt/cray/ssa/default/etc/cray-ssa`. It is symbolically linked from `/etc/cron.d/cray-ssa`.

8. Validate the configuration.

```
MGMT0# ssacli --check_conf
[stdout] Configuration File and CLI Options Valid.
```

The `ssacli` command should only be executed on the primary management node. If output differs from the above, review the messages presented and make corrections to the configuration file.

9. Use `scp` to copy the configuration from the primary management node to the secondary management node.

```
MGMT0# scp -p /opt/cray/ssa/default/etc/shepherd.conf \
snx99999n001:/opt/cray/ssa/default/etc/shepherd.conf
```

9.2.2.2 Collect and Upload a Snapshot

Prerequisites

IMPORTANT: This procedure is not necessary if upgrading to a newer version of SSA software.

This procedure must be performed on the Sonexion primary management node as `root`. The output examples in this section have been reduced for the sake of brevity. They will vary depending on the task being performed.

About this task

SSA terminology:

Run Set Each (shepherd) plugin is associated with zero or more plugin run sets. A run set is an alpha-numeric, textual label for a configuration that the shepherd uses to select plugins for execution. If `ssacli` is not invoked with `--runset` options, the default run set is `default`. Plugins can, and often are, associated with multiple plugin run sets.

Output Channel Every shepherd collection and related snapshot is associated with exactly one output channel (channel). A channel is an alpha-numeric label. If `ssacli` is not invoked with the `--channel` option, the default channel is `default`. Channels allow information collected and reported by SSA to be categorized by use.

Scenarios Beginning with release 1.7.0 SSA for ClusterStor supports the **Scenarios** feature introduced in the Shepherd 1.6.3 release. The only scenario currently implemented for ClusterStor is `triage` which depecrates the need to supply both a `--runset` and `--channel` when collecting a triage snapshot. This simplifies the command to collect a triage snapshot to only `ssacli --scenario=triage`.

Release 1.7.0 contains four run sets, each targeting a different SSA use case. These run sets are provided in the table.

Table 1. Supported Shepherd Run Configurations on Sonexion Systems

Purpose	Scenario	Run Set	Output Channel	Scheduled/On-Demand	Frequency
Baselining and detecting change in product configurations	Not implemented	`default`	`default`	Either	Once daily
Product health monitoring	Not implemented	`health`	`health`	Scheduled	Every 15 minutes
Seagate telemetry information gathering	Not implemented	`seagate_telemetry`	`seagate_telemetry`	Scheduled	Every 15 minutes (offset by 7 minutes from `health`)
Product support bundle capture	`triage`	`triage`	`triage`	On-demand	As requested

To observe behavior for the `default` run set and channel, perform each stage of the shepherd process separately and review the shepherd output on the system console.

> **IMPORTANT:** This procedure is not necessary if upgrading to a newer version of SSA software.

Procedure

1. Execute a collection for the `default` run set and channel.

   The execution time can vary between several minutes to up to 20 minutes on the largest Sonexion systems (more than 100 SSUs).

```
[root@snx99999n000]# ssacli --collect
[stdout] UI master_control status is (enabled)
[stdout] MASTER CONTROLS -> (M:True, C:True, S:True, U:True)
[stdout] UI CONTROLS -> (C:True, S:False, U:False)
[stdout] Shepherd Session: 1515075962
[stdout] Exclusive run: Lock file created @ /mnt/mgmt/var/opt/cray/ssa/lock/ssa.lock_channel-
default_device-snx99999n000_SNX2000_99999
[stdout] COLLECT stage start
[stdout] PLOAD: 28 plugin source modules loaded
[stdout] Collection Session:   '1515075962'
[stdout] Collection Directory: '/mnt/mgmt/var/opt/cray/ssa/collection/snx99999n000_SNX2000_99999/
default/1515075962'
[stdout] Collection Channel:   'default'
[stdout] Run Sets:             '['default']'
[stdout] Explicit Plugins:     'None'
...
[stdout] ** Entering run-level 20 **
[stdout] Plugin 'shepherd.encode.diagnostic' started
[stdout] Plugin 'shepherd.encode.diagnostic' stopped, return 0, time 9.67
[stdout] 25333557.0 raw bytes collected via directives.
[stdout] no collection directories meet purge requirements
[stdout] COLLECT stage stop (normally)
[stdout] Collection output directory @ /mnt/mgmt/var/opt/cray/ssa/collection/
snx99999n000_SNX2000_99999/default/1515075962
[stdout] shepherd session stop successfully
[stdout] run took 100.70 seconds
```

If the `COLLECT` stage stops either normally or as stated with survivable exception, the collection process was successful. The shepherd creates a text report named `collection_report.txt` in the collection output directory located
in: `/mnt/mgmt/var/opt/cray/ssa/collection/<collection_device>/default/<timestamp>`

This report provides a high-level summary of the collection, including:

- Information on the shepherd
- Amount of storage consumed by the collection
- The status of health checks the shepherd performed during the collection
- Platform summary information
- A plugin execution summary trace

This report can be useful to local operators in reviewing system status and high-level configuration. Survivable exceptions are part of the shepherd design. Individual plugins can fail in controlled ways, report these failures, and continue operation.

2. Execute a snapshot for the `default` run set and channel.

```
[root@snx99999n000]# ssacli --snapshot
[stdout] UI master_control status is (enabled)
[stdout] MASTER CONTROLS -> (M:True, C:True, S:True, U:True)
[stdout] UI CONTROLS -> (C:False, S:True, U:False)
[stdout] Shepherd Session: 1515076208
[stdout] Exclusive run: Lock file created @ /mnt/mgmt/var/opt/cray/ssa/lock/ssa.lock_channel-
default_device-snx99999n000_SNX2000_99999
[stdout] Starting SNAPSHOT stage
[stdout] Added '/mnt/mgmt/var/opt/cray/ssa/collection/snx99999n000_SNX2000_99999/default/1515075962'
to snapshot source list
[stdout] Est 18130548 bytes needed to snapshot, based on raw storage of 1 collection(s)
[stdout] Snapshot encoding dir created at '/mnt/mgmt/var/opt/cray/ssa/snapshot/default/isodx/staq01/
Linux/out/snx99999n000_SNX2000_99999/1515075962'
[stdout] no snapshot directories meet purge requirements
[stdout] Stopping SNAPSHOT stage normally
[stdout] shepherd session stop successfully
[stdout] run took 6.75 seconds
```

The `SNAPSHOT` stage should complete normally. Report other status messages to Cray support.

3. Invoke an upload of the `default` run set and channel.

```
[root@snx99999n000]# ssacli --upload
[stdout] UI master_control status is (enabled)
[stdout] MASTER CONTROLS -> (M:True, C:True, S:True, U:True)
[stdout] UI CONTROLS -> (C:False, S:False, U:True)
[stdout] Shepherd Session: 1515076262
[stdout] Exclusive run: Lock file created @ /mnt/mgmt/var/opt/cray/ssa/lock/ssa.lock_channel-
default_device-snx99999n000_SNX2000_99999
[stdout] Starting UPLOAD stage
[stdout] Upload Organization: staq01
[stdout] Upload Server: 136.162.62.191
[stdout] Upload Device: snx99999n000_SNX2000_99999
[stdout] Stopping UPLOAD stage normally
[stdout] shepherd session stop successfully
[stdout] run took 14.16 seconds
```

The `UPLOAD` stage should complete successfully. If the stage does not complete successfully, ensure the connectivity requirement has been met. Then report the issue to Cray support.

If the process above completes successfully, the first snapshot of the system support information is uploaded to Cray.

> **NOTE:** Cray recommends that the `triage` step be completed on an initial install of SSA to make sure there are no issues when collecting a triage snapshot.

4. Optional: If this is an initial installation, repeat steps 1 through 3 for the `triage` run set and channel. Add command line `--scenario=triage` to each of the `ssacli` command lines for `--collect`, `--snapshot`, and `--upload`.

9.3 ClusterStor InfiniBand Fabric Plugins

The 1.7.0 ClusterStor SSA release includes two plugins to collect and reset InfiniBand error counters. Error counters that are collected and reset are from all nodes that are connected to the ClusterStor InfiniBand fabric and logged into the subnet manager (SM). The plugin to collect the error counters is executed as part of the SSA triage,default, and health run sets. The health run set will be executed every 15 minutes. As of the 1.6.0 release, the plugin to reset the InfiniBand error counters (`cluster.network.infiniband.faberrcnt.reset plugin`) is disabled by default.

Sites that have no other mechanism of clearing the Infiniband error counters should enable the `cluster.network.infiniband.faberrcnt.reset` plugin. Not clearing the Infiniband error counters on a regular basis will result in erroneous data being collected.

For the procedure to disable the `cluster.network.infiniband.faberrcnt.reset` plugin, please refer to Cray *SFDC Article 6454*, *SSA Sonexion: SSA Sonexion 1.6.0 InfiniBand fabric enable reset plugin*.

10 Configure LNet Fine-Grained Routing for XC Systems

Tasks for Configuring Fine-grained Routing

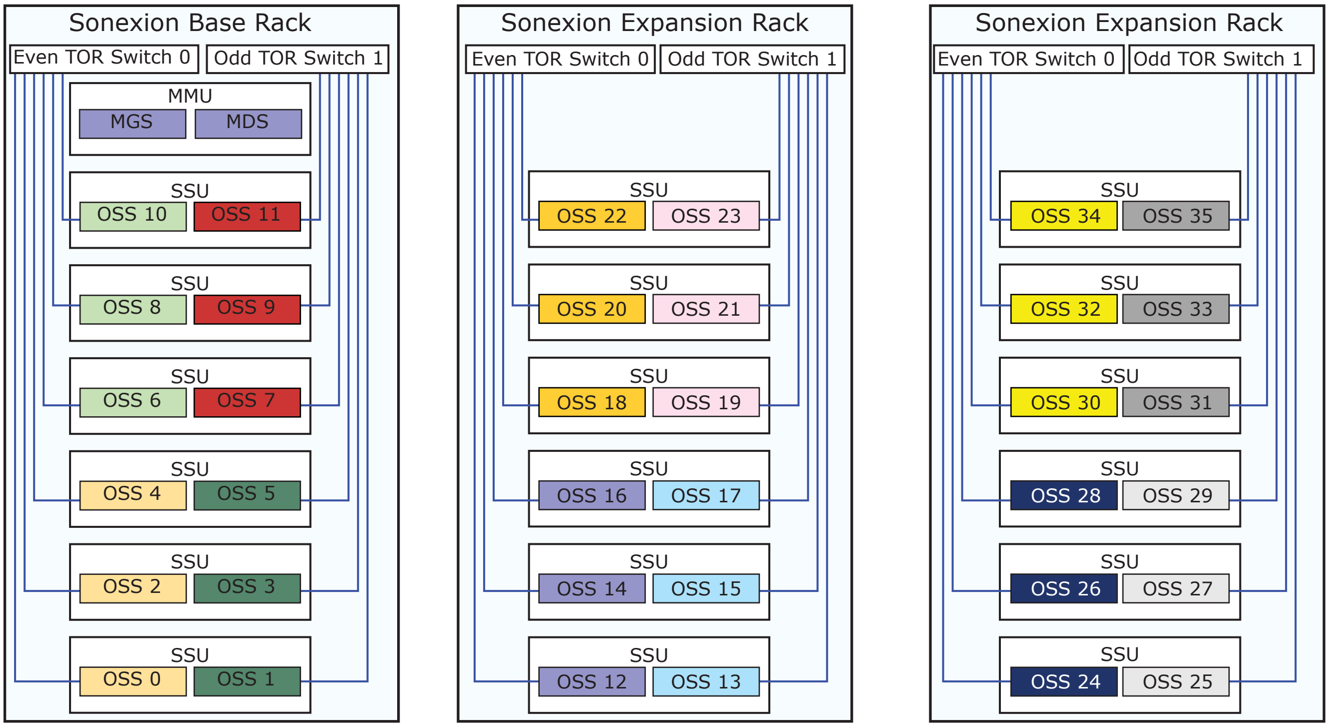
The following tasks must be completed to configure fine-grained routing (FGR) on Cray and Sonexion systems:

- Use bandwidth matching to get the router-to-server ratio
- Determine the IP addressing scheme to support the system configuration on the Cray and Sonexion systems
- Use CLCVT to generate the LNet configuration (`lnet.conf`, `routes.conf`, and `ip2nets.conf`)
- Configure IP addressing on IB interfaces (IPoIB)
- Place the LNet configuration the CLE system
- Place LNet configuration on the Sonexion
- Verify and test the configuration

Sonexion File Systems

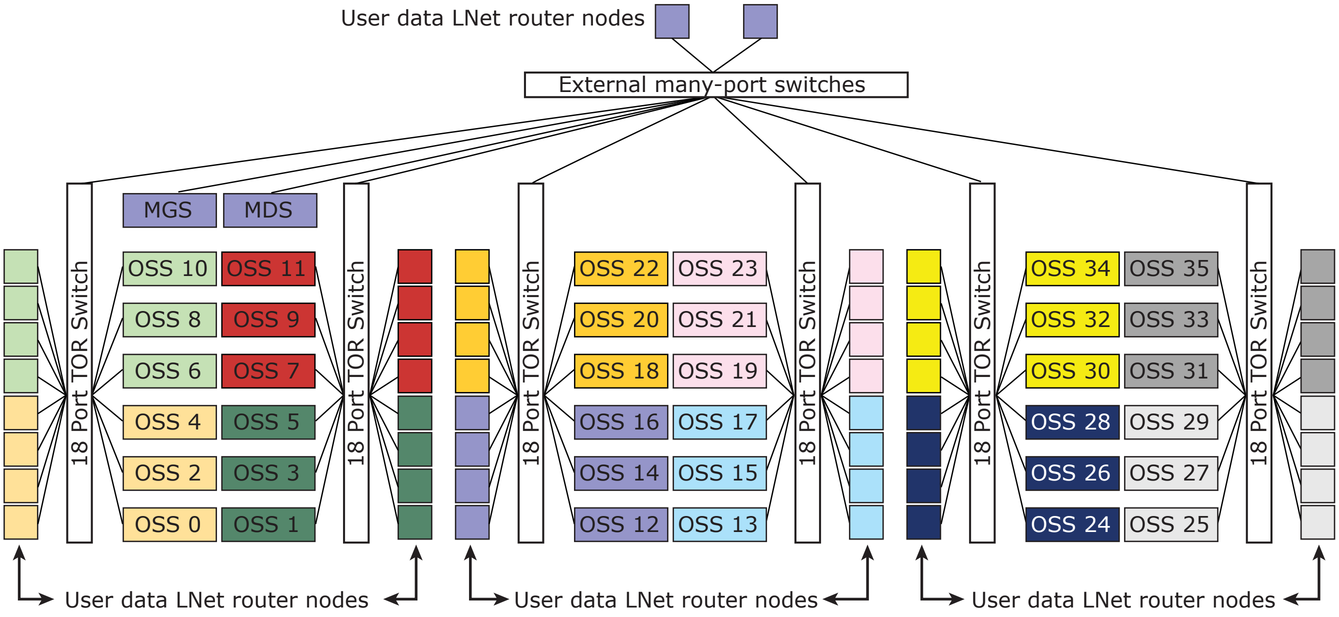
Sonexion file systems are made of two basic building blocks: scalable storage units (SSUs) and a metadata management unit (MMU). Each SSU houses disk drives and two OSSs that are connected to an even or odd top-of-rack IB switch and then to the system storage network. The MMU houses the MDS/MGS server pair and each file system includes an MDS/MGS server pair.

Figure 14. Sonexion File System Components



LNet FGR 3-Rack Example on page 61 shows how FGR is configured for three Sonexion racks. Each color represents an LNet group that is optimized to match the maximum bandwidth capacity for a Sonexion OSS. Four Cray LNet router nodes support three Sonexion OSSs in this example.

Figure 15. LNet FGR 3-Rack Example



InfiniBand Infrastructure

Cray uses industry standard fourteen data rate (FDR) InfiniBand infrastructure (although LNet supports other HSNs). The FDR InfiniBand network infrastructure (IBNI) is typically composed of dual InfiniBand core switches, and redundant pathways. There are two independent paths for all functionality accessed through the IBNI. This is accomplished with the use of active-active pairing for every function. In the event of an IBNI (or node) failure removing access to a node, the active partner(s) will automatically take over operation. Once the failure is resolved, the failed node is placed back into service. All components can be serviced, to include removal/ replacement, in the event of a single failure without loss of access to the file system.

InfiniBand subnet routing must be implemented that does not result in credit loops, (deadlock avoidance) and that high-bandwidth connections take paths through the subnet to avoid bandwidth conflicts. Connections from the HPC systems are typically split, to allow maximum bandwidth in the event of a total switch failure.

Lustre LNet routers can connect two distinct InfiniBand fabrics. Any number of LNet routers may be used in parallel to achieve the desired performance level. The hardware to create such an LNet router can be as simple as a 1U rackmount server with two InfiniBand HCA interfaces and standard Lustre software.

XC Series Client Interface

The connection from the XC Series system to the Sonexion is accomplished with a set of LNet router nodes attached to the internal Aries® HSN. LNet routers typically have FDR InfiniBand connections to director class switches (evenly split between the two). For any given file system, the Lustre clients use all appropriate and available LNet routers in a round-robin fashion. The effect of this is to add together the performance of each useable LNet router for a total available bandwidth number. The LNet routers demonstrate nearly linear performance scaling as they act in parallel.

A single XC Series I/O blade supports two independent I/O nodes that each run the Cray Linux Environment (CLE). Each I/O node supports two PCIe Gen3 x8 slots, each can support single or dual-port IB HCAs, to provide two compute node LNet routers.

A single port reaches the maximum available PCIe bandwidth. Each HCA is capable of approximately 6GB/s of raw IB data transfer. In the LNet router configuration, this translates into approximately 5.5GB/s of Lustre data traffic. However—due to interference between the two HCAs and the limits of the HSN interface—when both HCAs are active with Lustre traffic, the total is 8.3GB/s (or 4.15GB/s for each of the two IB links).

> **IMPORTANT:** Due to routing restrictions imposed by both Lustre LNet and the Internet Protocol (IP), which provides the underlying node addressing, each of the two HCA ports must be assigned to a different IP subnet and LNet o2ib instance number.

In addition, the IP subnet must be different for the HSN interface. This allows the LNet router to distinguish the appropriate exit path for any packet coming into the router. The IBNI typically includes dual-core switches and is designed to maintain the highest possible performance during any failure. Therefore, each of the two HCAs on a LNet router node are connected to different core switches. When a switch fails, the normal 4.15GB/s available to each port can become 5.5GB/s on the port surviving the failure.

The address assignments for each file system must use unique subnets. Therefore, any given LNet router IB link services only the file system that matches the router-assigned subnet. Because the two links on a single LNet router node must have different subnets, it follows that each LNet router node services two different file systems.

Metadata Server Restrictions

Lustre metadata servers (MDS/MGS) require special consideration. The traffic for metadata is very latency sensitive, but does not use much bandwidth. If this traffic were to be interleaved with normal read/write data traffic, it would be subject to the transmission delays imposed by large data transfers. These are redundant connections, and either of the two links for a single file system is sufficient to support operations. Each of the two links for a single file system is connected to a different core switch, allowing operations to proceed in the event of either a LNet router or a switch failure.

Distributing links across I/O blades accommodates system service procedures. If an LNet router node fails, the entire I/O blade may be serviced to resolve the problem. Placement of I/O blades purposed as LNet routers is distributed across different cabinets so that any failure affecting an entire cabinet will only take a single pair of LNet router nodes out of service.

10.1 Routing and Bandwidth Matching for Sonexion Systems

Cray recommends two IB links per router node FDR InfiniBand links on XC Series systems.

Sonexion bandwidth is evenly spread across all OSS nodes. Peak usable bandwidth is 2.4GB/s per OSS. External connections to the TOR switches must allow for this peak plus additional overhead, for a total of 3.3GB/s per OSS. A single InfiniBand FDR link can carry two OSSs worth of traffic (nominally 6.8GB/s). It is desirable to have an even number of links per TOR switch. The bandwidth of the HSN LNet nodes are matched to the bandwidth of the of the OSS nodes on the IB network for optimal throughput.

> **RESTRICTION:** When two IB interfaces are used for each LNet router node, each IB interface must be on separate IPv4 subnets.

Sonexion storage systems are composed of two basic building blocks—a single Metadata Management Unit (MMU) and one or more Scalable Storage Units (SSUs). An optional building block, the additional distributed namespace environment (DNE) unit (ADU), is also available. The MMU consists of two management servers (MGS) and two metadata servers (MDSs) and either a 2U24 or 5U84 drive enclosure. An SSU consists of two OSSs with a 5U84 drive enclosure.

An ADU consists of two MDSs with a 2U24 drive enclosure. The Sonexion 1600 MDRAID and GridRAID systems provide 5GB/s per SSU sustained, and 6GB/s per SSU peak. The Sonexion 2000 system provides 7.5GB/s per SSU sustained, and 9 GB/s per SSU peak. Ensuring sufficient network bandwidth to each OSS is a key requirement in the design of Lustre networks.

An XC Series I/O blade has one Aries Network Interface Controller (NIC) that provides an I/O module (IBB) with 17GB/s of I/O bandwidth. That IBB includes two nodes (LNet routers), so each node can achieve 8.5GB/s. Each LNet router node can support one or two, single- or dual-port IB host channel adapters (HCAs). Each active HCA port (`ib0` and `ib2`, for instance) must be assigned to a different LNet (cannot bond them to a single LNet). Therefore, a single LNet router node services two different LNets. A single FDR (Fourteen Data Rate) IB HCA is capable of 5.5GB/s of LNet traffic. Therefore, if busy, the two IB HCAs on a single LNet router split 8.5GB/s and achieve 4.25GB/s per IB port.

Because a single FDR IB link provides sufficient bandwidth for a single Sonexion 2000 OSS, the ratio of *n* IB links to *n* servers would work. However, in the case of the Sonexion 1600, this results in wasted bandwidth. *Bandwidth Capabilities of 6 Sonexion 2000 OSSs (3 SSUs) (22.5GB/s) 5 IB Links from Single HCA Routers (27.50GB/s)* on page 64 indicates that six Sonexion 1600 OSSs can deliver 18GB/s peak I/O bandwidth. If six single HCA LNet router links are assigned to service the six servers, then the fabric is capable of 33GB/s. This is nearly double the bandwidth that six OSSs can deliver. If IB links from LNet routers with dual HCAs are assigned, then the configuration would provide 23% more bandwidth than what is needed.

Assigning two single HCA LNet router links (or three IB links from dual HCA routers) to every four Sonexion-1600 servers is ideal from the perspective of minimizing wasted bandwidth. This would either result, however, in FGR groups that span more than one TOR switch or would require additional sub-optimal FGR groups that contain just two servers. If only FGR groups are considered (where the number of servers in each FGR group is evenly divisible by the number of servers in each TOR switch) then this restricts the configurations to ratios with one, two, three or six servers.

With the above constraints in mind, suitable ratios are:

- Sonexion 1600
  - XC 40 Single HCA: 2:3
  - XC 40 Dual HCA: 5:6, or n:n+1
- Sonexion 2000
  - XC 40 Single HCA: n:n
  - XC 40 Dual HCA: n:n

Sonexion 2000 Bandwidth Matching

The bolded table cells in each table show the best grouping of Cray XC Series LNet nodes and Sonexion OSS nodes over single or dual InfiniBand HCAs. Always assign more bandwidth in LNet router nodes than is provided by the disk I/O to/from the OSSs.

- Based on the IOR benchmark with Cray chosen parameters and read/write averaging per SSU, the bandwidth is 6.0 GB/sec per SSU (3.0GB/s per OSS) for 2TB drives and requires an LNet ratio of 1:1
- Based on the IOR benchmark with Cray chosen parameters and read/write averaging per SSU, without an ESU, the bandwidth is 7.5GB/s per SSU (3.75GB/s per OSS) for 4TB and 6TB drives and requires an LNet ratio of 1:1
- Based on the IOR benchmark with Cray chosen parameters and read/write averaging per SSU, with an ESU, the bandwidth is 9.0 GB/sec per SSU (4.5GB/s per OSS) for 4TB and 6TB drives and requires an LNet ratio of 7:6

Table 2. Bandwidth Capabilities of 6 Sonexion 2000 OSSs (3 SSUs) (22.5GB/s) 5 IB Links from Single HCA Routers (27.50GB/s)

	1	2	3	4	5	6
Sonexion 1600 OSS	3.00	6.00	9.00	12.00	15.00	18.00
Sonexion 2000 OSS	3.75	7.50	11.25	15.00	18.75	***22.50***
Single HCA	5.50	11.00	16.50	22.00	***27.50***	33.00
Dual HCA	4.20	8.40	12.60	16.80	21.00	25.20

Table 3. 6 Sonexion 2000 OSSs (3 SSUs) (22.5GB/s) 6 IB Links from Dual-HCA Routers (25.20GB/s)

	1	2	3	4	5	6
Sonexion 1600 OSS	3.00	6.00	9.00	12.00	15.00	18.00
Sonexion 2000 OSS	3.75	7.50	11.25	15.00	18.75	***22.50***
Single HCA	5.50	11.00	16.50	22.00	27.50	33.00

	1	2	3	4	5	6
Dual HCA	4.20	8.40	12.60	16.80	21.00	***25.20***

Sonexion 3000 Bandwidth Matching

SSU Performance for IB with EDR LDN - Based on the IOR benchmark with Cray parameters and read/write averaging:

- 9GB/s per SSU for all 7200 RPM drive sizes (4.5GB/s per OSS)
- 10GB/s per SSU with an ESU, for all drive types (5GB/s per OSS)
- 12GB/s per SSU with 10K RPM HPC drives, (6GB/s per OSS)

The following table shows the number of physical connections (i.e. cables) required for connecting the TOR switches in each rack to a XC system.

Table 4. Number of Physical Connections Required for Sonexion 3000 LNet Routing

Number of SSUs	Single FDR HCA LNet Connections	Dual FDR HCA LNet Connections	Single EDR HCA Connections
7.2 RPM Drives			
1	1	2	1
2	2	3	2
3	3	4	2
4	4	5	3
5	5	6	3
6	6	7	4
7	7	8	4
10K HPC Drives			
1	2	2	1
2	3	3	2
3	4	4	2
4	5	6	3
5	6	7	4
6	7	8	4
7	8	10	5

10.2 External Server Node: Sonexion 3000 or ClusterStor L300 System Recommended Parameters

For Sonexion 3000 or ClusterStor L300 systems, the `peer_credits` setting must be consistent across all InfiniBand (IB) peers on the Lustre network. When routers and/or external Lustre clients have mlx5-based Host Channel Adapters (HCAs), `map_on_demand` must be set to 0. In addition, Cray recommends `peer_credits` and `concurrent_sends` be set to 16. Thus, when connecting to a Sonexion 3000/ClusterStor L300, the recommended parameters are generally the same as those for the Sonexion 900, 1600, and 2000—except that `map_on_demand` must be set to 0, and `peer_credits` and `concurrent_sends` should be set to 16 for all IB peers on the Lustre network.

If an IB peer must have access to an mlx4-based file system (i.e. Sonexion 900, Sonexion 1600, and Sonexion 2000) and an mlx5-based file system (i.e. Sonexion 3000), the `ko2iblnd` parameters of all mlx4 peers must match the `ko2iblnd` mlx5-peer parameters to ensure shared mlx4- and mlx5-peer function. For example, in a system where an external Login node needs access to a Sonexion 2000 and Sonexion 3000, all mlx4- and mlx5-peer `ko2iblnd` parameters should match the LNet parameters recommended for a Sonexion 3000.

For systems that have a Sonexion 3000/ClusterStor L300 running at running software version 2.1-SU003 or greater, `peer_credits` can be increased to 84 and `concurrent_sends` can be increased to 42. All IB peers within the Lustre network must be able to support these same values if they are to be used.

Cray recommends an object-based disk (OBD) timeout of 100 seconds, which is the default value. Set this parameter using the `lctl conf_param` command on the management server (MGS). For example:

```
$ lctl conf_param fs_name.sys.timeout=100
$ cat /proc/sys/lustre/timeout
100
```

`ko2iblnd:timeout`

Default **10**. The `o2iblnd` timeout in seconds. Cray recommends setting this to 10 seconds.

`ko2iblnd:peer_timeout`

Default **0**. Number of seconds without aliveness news it takes to declare a peer dead. Cray recommends setting this to 0.

`ko2iblnd:keepalive`

Default **30**. Idle time in seconds before sending a keepalive. Cray recommends setting this to 30.

`ko2iblnd:credits`

Default **2048**. Number of concurrent sends allowed by `o2iblnd`. Shared by all CPU partitions (CPT). Cray recommends setting this to 2048.

`ko2iblnd:ntx`

Default **2048**. Number of message descriptors allocated for each pool. Cray recommends setting this to 2048.

`ko2iblnd:peer_credits` 16

Enter the value for the `ko2iblnd` parameter `peer_credits`. This is the number of concurrent sends to a single peer. This value must be the same on all external login clients and the Lustre file system servers.

`ko2iblnd:concurrent_sends` 16

Determines send work queue sizing. If this option is omitted, the default is calculated based on the values of `peer_credits` and `map_on_demand`. This value must be the same on the external login clients and the Lustre file system servers.

ko2iblnd:map_on_demand

Default **0**. Controls the use of fast memory registration (FMR). Cray recommends setting this value to 0 for InfiniBand HCAs or a value of 32 for Intel® Omni-Path (OPA) host fabric interfaces (HFI).

lnet:router_ping_timeout

Default **50**. Number of seconds to wait for the reply to a router health query. Cray recommends using the default value of 50 seconds.

lnet:live_router_check_interval

Default **60**. Number of seconds between live router health checks. Cray recommends leaving this at the default value of 60 seconds. A value less than or equal to 0 disables pinging of live routes.

lnet:dead_router_check_interval

Default **60**. Number of seconds between dead router health checks. Cray recommends using the default value of 60 seconds. A value less than or equal to 0 disables pinging of dead routes.

lnet:avoid_asym_router_failure 1

Avoid asymmetrical router failures (0 to disable; 1 to enable).

10.3 Use CLCVT to Configure Fine-Grained Routing Files

The clcvt command, available on the boot node and the system management workstation (SMW), aids in the configuration of Lustre networking (LNet) fine-grained routing (FGR).

The clcvt command requires several file-system-specific input files and generates LNet kernel module configuration information that can be used to configure the servers, routers, and clients for that file system. The utility can also create cable maps (in HTML, CSV, and human-readable formats) and validate cable connection on Sonexion systems.

See the clcvt(8) man page for detailed information.

10.3.1 CLCVT Prerequisite Files

The clcvt command requires several prerequisite files in order to compute the ip2nets and routes information for the specific configuration. The prerequisite files must be placed in an empty directory on the boot node or SMW, depending on where clcvt will be run.

Deciding how to assign which routers to which object storage servers (OSSs), what fine grained routing (FGR) ratios to use, which interface on which router to use for a Lustre networking (LNet) group, and router placement are all things that can vary greatly from site to site. LNet configuration is determined as the system is ordered and configured. See a Cray representative for the site-specific values. Use *Routing and Bandwidth Matching for Sonexion Systems* on page 63 as a guide.

info.*file-system-identifier*

A file with global file system information for the *cluster-name* server machine and each client system that will access it.

***client-system*.hosts**

A file that maps the client system (such as the Cray mainframe) IP addresses to unique host names, such as the boot node /etc/hosts file.

The *client-system* name must match one of the clients in the info.*file-system-identifier* file.

***client-system*.ib** A file that maps the client system LNet router InfiniBand IP addresses to system hardware cnames. The *client-system* name must match one of the clients in the info.*file-system-identifier* file. This file must be created by an administrator.

***clustername*.ib** A file that maps the Lustre server InfiniBand IP addresses to cluster (for example, Sonexion) host names. The *clustername* name must match the clustername in the info.*file-system-identifier* file. This file must be created by an administrator.

***client-system*.rtrIm** A file that contains rtr -Im command output (executed on the SMW) for the *client-system*.

10.3.2 Create the CLCVT info.file-system-identifier File

Create the info.*file-system-identifier* file manually. This file contains global file system information for the Lustre server machine and each client system that must have access. Based on the ratio of OSS to LNet routers in the configuration, the [*clustername*] section (snx11029n in this example) and each [*client-system*] section (hera in this example) defines which servers and routers belong to each InfiniBand (IB) subnet.

This file is in the form of a ini style file, and the possible keywords in the [info] section include clustername, ssu_count, and clients. Refer to the clcvt man page for detailed information and usage.

clustername Defines the base name used for all file system servers. The example show a Sonexion file system snx11029n. Thus, all server hostnames will be snx11029n*NNN*. *NNN* is a three-digit number starting at 000 and 001 for the primary and secondary Cray Sonexion management servers (MGMT), 002 for the MGS, 003 for the MDS, 004 for the first OSS, and counting up from there for all remaining OSSs.

ssu_count Defines how many SSUs make up a Sonexion file system. If this is missing, then this is not a Sonexion file system but a CLFS installation.

clients Defines a comma-separated list of mainframe names that front-end this file system (hera in this example).

fgr_ratio Determines the Fine-Grained Routing ratio (see *Routing and Bandwidth Matching for Sonexion Systems* on page 63. Set this to *M*:*N*, where *M* is the number of routers per group and *N* is the number of servers per group. If set, oss_per_switch and ssu_per_rack must be equal to or greater than *N*, and they must be set to a whole-number multiple of *N*. Changing the ssu_per_rack and oss_per_switch default settings can accommodate seven SSUs per rack.

```
ssu_per_rack = 7
oss_per_switch = 7
```

The info.*file-system-identifier* file requires a [*client-system*] section for each client system listed in the clients line of the [info] section to describe the client systems and a [*clustername*] section to describe the Lustre server system.

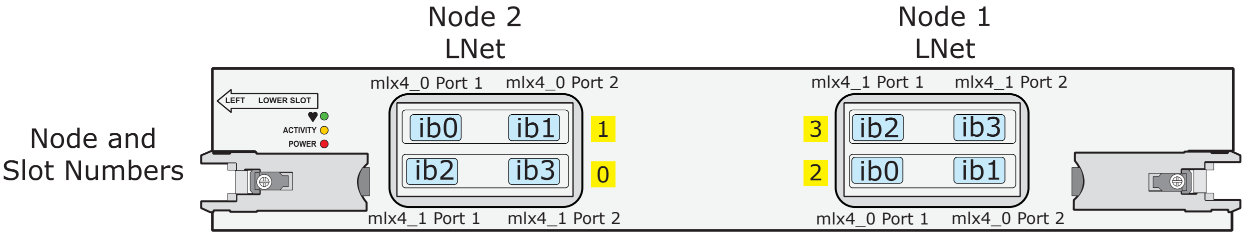
Each of these sections contain a literal LNet network wildcard in the format of *LNET-name*:*IP-wildcard* which instructs the LNet module to match a host IP address to *IP-wildcard* and, if it matches, instantiate LNet *LNET-name* on them.

The hostname fields in the [*client-system*] section of this file are fully qualified interface specifications of the form *hostname*(ib*n*), where (ib0) is the assumed default if not specified.

XC systems support multiple IB interfaces per LNet router. Configure the second IB interface and append the interface names (ib*n*) to the cname for the routers (as shown in the example). These interface names must also be appended to the *client-system*.ib file.

XC40 IB port assignments are shown in the illustration.

Figure 16. XC40 IB Port Assignments



Example: info.*file-system-identifier* File

This example configures an info.snx11029n file that supports the example file system in *XC40 with Two FDR IB Interfaces Per LNet Router* on page 71. It is configured with four FDR IB interfaces (two per router node) from each XC40 LNet router blade and includes a total of 12 SSUs for the file system.

The Sonexion OSSs that connect to the ib2 interface of the LNet routers in this example must have an IP address on the ib2 subnet. The ib0:1 alias is added as label for the ib2 interfaces because there is only single IB interface on each Sonexion OSS.

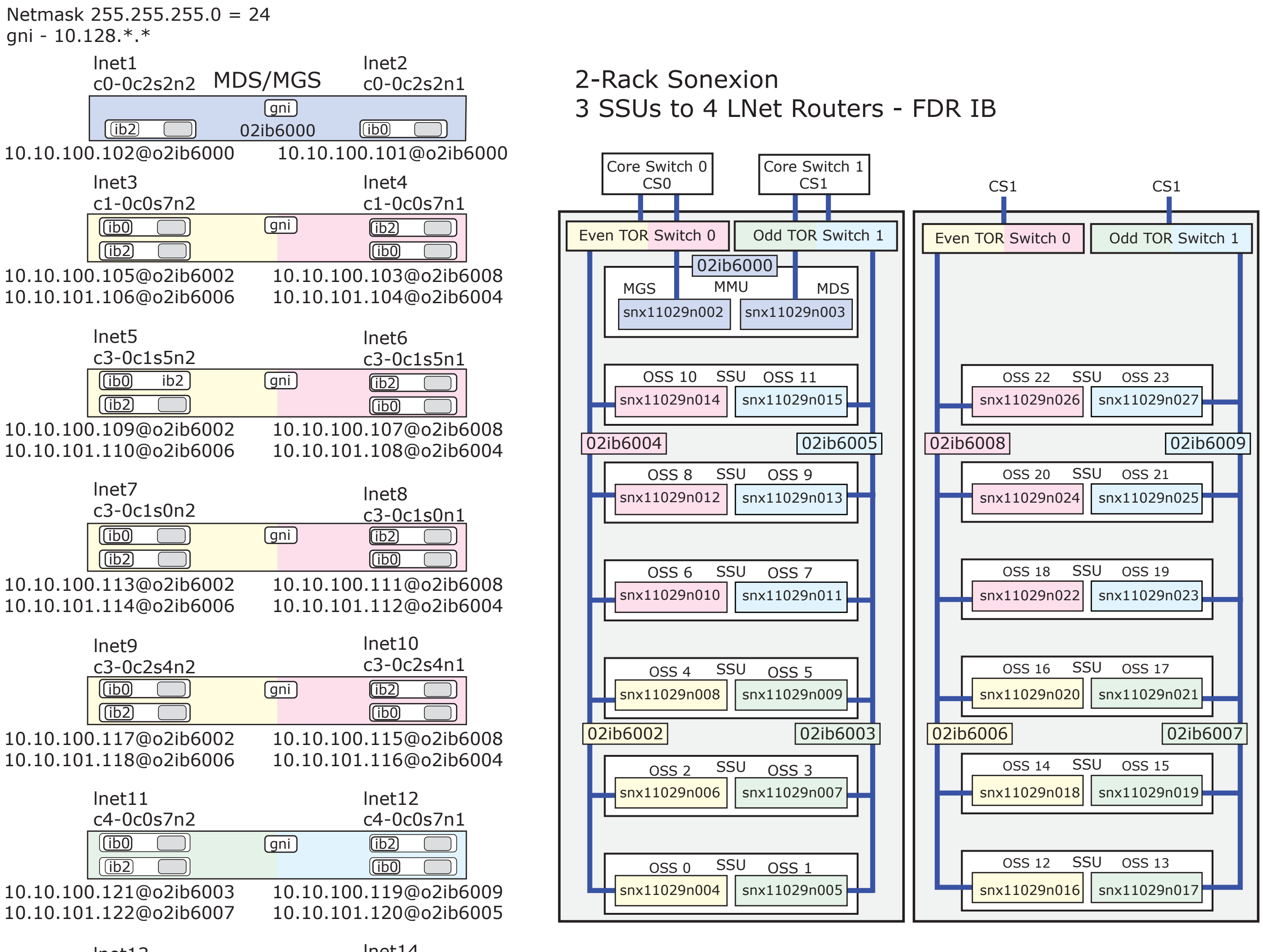
```
# This section describes the size of this filesystem.
[info]
clustername = snx11029n
SSU_count = 12
clients = hera
fgr_ratio = 4:3
# other optional keywords not used in this example
# oss_count
# oss_per_switch
# sonexion
# ssu_per_rack
# fgr

[hera]
lnet_network_wildcard = gni1:10.128.*.*

# Because of our cabling assumptions and naming conventions, we only
# need to know which XIO nodes are assigned to which LNETs.  From that
# our tool can actually generate a "cable map" for the installation folks.
o2ib6000: c0-0c2s2n2(ib2), c0-0c2s2n1(ib0) ; MGS and MDS
o2ib6002: c1-0c0s7n2(ib0), c3-0c1s5n2(ib0), c3-0c1s0n2(ib0), c3-0c2s4n2(ib0) ; OSSs 0/2/4
o2ib6003: c4-0c0s7n2(ib0), c5-0c1s5n2(ib0), c5-0c1s0n2(ib0), c5-0c2s4n2(ib0) ; OSSs 1/3/5
o2ib6004: c1-0c0s7n1(ib0), c3-0c1s5n1(ib0), c3-0c1s0n1(ib0), c3-0c2s4n1(ib0) ; OSSs 6/8/10
o2ib6005: c4-0c0s7n1(ib0), c5-0c1s5n1(ib0), c5-0c1s0n1(ib0), c5-0c2s4n1(ib0) ; OSSs 7/9/11
o2ib6006: c1-0c0s7n2(ib2), c3-0c1s5n2(ib2), c3-0c1s0n2(ib2), c3-0c2s4n2(ib2) ; OSSs 12/14/16
o2ib6007: c4-0c0s7n2(ib2), c5-0c1s5n2(ib2), c5-0c1s0n2(ib2), c5-0c2s4n2(ib2) ; OSSs 13/15/17
o2ib6008: c1-0c0s7n1(ib2), c3-0c1s5n1(ib2), c3-0c1s0n1(ib2), c3-0c2s4n1(ib2) ; OSSs 18/20/22
o2ib6009: c4-0c0s7n1(ib2), c5-0c1s5n1(ib2), c5-0c1s0n1(ib2), c5-0c2s4n1(ib2) ; OSSs 19/21/23

[snx11029n]
lnet_network_wildcard = o2ib0:10.10.100.*
```

```
o2ib6000: snx11029n002(ib0), snx11029n003(ib0) ; MGS and MDS
o2ib6002: snx11029n004(ib0), snx11029n006(ib0), snx11029n008(ib0) ; OSSs 0/2/4
o2ib6003: snx11029n005(ib0), snx11029n007(ib0), snx11029n009(ib0) ; OSSs 1/3/5
o2ib6004: snx11029n010(ib0), snx11029n012(ib0), snx11029n014(ib0) ; OSSs 6/8/10
o2ib6005: snx11029n011(ib0), snx11029n013(ib0), snx11029n015(ib0) ; OSSs 7/9/11
o2ib6006: snx11029n016(ib0:1), snx11029n018(ib0:1), snx11029n020(ib0:1) ; OSS 12/14/16
o2ib6007: snx11029n017(ib0:1), snx11029n019(ib0:1), snx11029n021(ib0:1) ; OSS 13/15/17
o2ib6008: snx11029n022(ib0:1), snx11029n024(ib0:1), snx11029n026(ib0:1) ; OSS 18/20/22
o2ib6009: snx11029n023(ib0:1), snx11029n025(ib0:1), snx11029n027(ib0:1) ; OSS 19/21/23
```

The example shows two interfaces for the MDS/MGS nodes on a single blade. If the metadata server must be on a 3rd IPv4 subnet, then the additional interfaces would be placed on other LNet nodes.

Move the `info.snx11029n` file to the working directory where the `clcvt` command will be run.

Figure 17. XC40 with Two FDR IB Interfaces Per LNet Router



snx11029n002	MGS	ib0	10.10.100.3	o2ib6000
snx11029n003	MDS	ib0	10.10.100.4	
snx11029n004	OSS0	ib0	10.10.100.5	02ib6002
		ib0	10.10.101.5(ib0:1)	
snx11029n005	OSS1	ib0	10.10.100.6	02ib6003
		ib0	10.10.101.6(ib0:1)	
snx11029n006	OSS2	ib0	10.10.100.7	o2ib6002
		ib0	10.10.101.7(ib0:1)	
snx11029n007	OSS3	ib0	10.10.100.8	o2ib6003
		ib0	10.10.101.8(ib0:1)	
•				
•				
•				
snx11029n027	OSS23	ib0	10.10.100.26	o2ib6009
		ib0	10.10.101.26(ib0:1)	

10.3.3 Create the CLCVT client-system.hosts File

Use the `/etc/hosts` file from the XC boot node to create the *`client-system`*`.hosts` file (`hera.hosts`). Copy the `/etc/hosts` file from the boot node to a working directory and include the LNet router nodes. A typical `/etc/hosts` file is shown below.

Example *client-system*.hosts File

```
#
# hosts         This file describes a number of hostname-to-address
#               mappings for the TCP/IP subsystem.  It is mostly
#               used at boot time, when no name servers are running.
#               On small systems, this file can be used instead of a
#               "named" name server.
# Syntax:
#
# IP-Address  Full-Qualified-Hostname  Short-Hostname
#

127.0.0.1       localhost

# special IPv6 addresses
::1     ipv6-localhost  localhost       ipv6-loopback

fe00::0 ipv6-localnet

ff00::0 ipv6-mcastprefix
ff02::1 ipv6-allnodes
ff02::2 ipv6-allrouters
ff02::3 ipv6-allhosts

10.128.0.1      nid00000        c0-0c0s0n0      dvs-0
10.128.0.2      nid00001        c0-0c0s0n1      boot001 boot002
10.128.0.31     nid00030        c0-0c0s0n2      #ddn_mds
10.128.0.32     nid00031        c0-0c0s0n3      hera-rsip2
10.128.0.3      nid00002        c0-0c2s2n1      lnet2
10.128.0.4      nid00003        c0-0c2s2n2      lnet1
10.128.0.29     nid00018        c1-0c0s7n1      lnet4
10.128.0.30     nid00019        c1-0c0s7n2      lnet3
10.128.0.45     nid00024        c3-0c1s0n1      lnet8
10.128.0.46     nid00025        c3-0c1s0n2      lnet7
10.128.0.57     nid00036        c3-0c1s5n1      lnet6
10.128.0.58     nid00037        c3-0c1s5n2      lnet5
10.128.0.67     nid00046        c3-0c2s4n1      lnet10
10.128.0.68     nid00047        c3-0c2s4n2      lnet9
10.128.0.75     nid00054        c4-0c0s7n1      lnet12
10.128.0.76     nid00055        c4-0c0s7n2      lnet11
10.128.0.85     nid00067        c5-0c1s5n1      lnet14
10.128.0.86     nid00068        c5-0c1s5n2      lnet13
10.128.0.91     nid00074        c5-0c1s0n1      lnet16
10.128.0.92     nid00075        c5-0c1s0n2      lnet15
10.128.0.101    nid00093        c5-0c2s4n1      lnet18
10.128.0.102    nid00094        c5-0c2s4n2      lnet17
. . .
```

Move the *`client-system`*`.hosts` file to the working directory on the SMW or boot node where the `clcvt` command will be run.

10.3.4 Create the CLCVT client-system.ib File

The *`client-system`*`.ib` file contains a client-system LNet router InfiniBand (IB) IP address to `cname` mapping information in an `/etc/hosts` style format. The hostname field in this file is a fully-qualified interface specification of the form *`hostname`*`(ibn)`, where `(ib0)` is the assumed default if not specified. This file must be created by an administrator.

XC systems can support multiple IB interfaces per router—configure the second IB interface and append the interface names (ib*n*) to the cname for the routers. The LNet router IB IP addresses should be within the same subnet as the Lustre servers (MGS/MDS/OSS). One possible address assignment scheme would be to use a contiguous set of IP addresses, with ib0 and ib2 on each node having adjacent addresses. These interface names must be appended to the info.*file-system-identifier* file.

Example: *client-system*.ib File

This example (hera.ib) configures two FDR IB cards per XC system LNet router.

```
#
# This is the /etc/hosts-like file for Infiniband IP addresses
# on "hera".
#
10.10.100.101    c0-0c2s2n1(ib0)
10.10.100.102    c0-0c2s2n2(ib2)
10.10.100.103    c1-0c0s7n1(ib2)
10.10.101.104    c1-0c0s7n1(ib0)
10.10.100.105    c1-0c0s7n2(ib0)
10.10.101.106    c1-0c0s7n2(ib2)
10.10.100.107    c3-0c1s5n1(ib2)
10.10.101.108    c3-0c1s5n1(ib0)
10.10.100.109    c3-0c1s5n2(ib0)
10.10.101.110    c3-0c1s5n2(ib2)
10.10.100.111    c3-0c1s0n1(ib2)
10.10.101.112    c3-0c1s0n1(ib0)
10.10.100.113    c3-0c1s0n2(ib0)
10.10.101.114    c3-0c1s0n2(ib2)
10.10.100.115    c3-0c2s4n1(ib2)
10.10.101.116    c3-0c2s4n1(ib0)
10.10.100.117    c3-0c2s4n2(ib0)
10.10.101.118    c3-0c2s4n2(ib2)
10.10.100.119    c4-0c0s7n1(ib2)
10.10.101.120    c4-0c0s7n1(ib0)
10.10.100.121    c4-0c0s7n2(ib0)
10.10.101.122    c4-0c0s7n2(ib2)
10.10.100.123    c5-0c1s5n1(ib2)
10.10.101.124    c5-0c1s5n1(ib0)
10.10.100.125    c5-0c1s5n2(ib0)
10.10.101.126    c5-0c1s5n2(ib2)
10.10.100.127    c5-0c1s0n1(ib2)
10.10.101.128    c5-0c1s0n1(ib0)
10.10.100.129    c5-0c1s0n2(ib0)
10.10.101.130    c5-0c1s0n2(ib2)
10.10.100.131    c5-0c2s4n1(ib2)
10.10.101.132    c5-0c2s4n1(ib0)
10.10.100.133    c5-0c2s4n2(ib0)
10.10.101.134    c5-0c2s4n2(ib2)
```

Move the *client-system*.ib file to the working directory on the SMW or boot node where the clcvt command will be run.

10.3.5 Create the CLCVT cluster-name.ib File

The CLCVT *cluster-name*.ib file contains Lustre server InfiniBand (IB) IP addresses to cluster (Sonexion) host name mapping information in a /etc/hosts style format. This file must be created by an administrator.

The Sonexion servers that connect to the ib2 interface of the routers in this example must have an IP address on the ib2 subnet. The ib0:1 alias is added as a label for the ib2 interfaces to support the single IB interface on each Sonexion OSS.

Example *cluster-name*.ib File*snx11029n*.ib

```
#
# This is the /etc/hosts-like file for InfiniBand IP addresses
# on the Sonexion known as "snx11029n".
#
10.10.100.1    snx11029n000    #mgmnt
10.10.100.2    snx11029n001    #mgmnt
10.10.100.3    snx11029n002    #mgs
10.10.100.4    snx11029n003    #mds
10.10.100.5    snx11029n004    #first oss, oss0
10.10.100.6    snx11029n005    #oss1
10.10.100.7    snx11029n006    #oss2
10.10.100.8    snx11029n007    #oss3
10.10.100.9    snx11029n008    #oss4
10.10.100.10   snx11029n009    #oss5
10.10.100.11   snx11029n010    #oss6
10.10.100.12   snx11029n011    #oss7
10.10.100.13   snx11029n012    #oss8
10.10.100.14   snx11029n013    #oss9
10.10.100.15   snx11029n014    #oss10
10.10.100.16   snx11029n015    #oss11
10.10.101.17   snx11029n016(ib0:1)    #oss12
10.10.101.18   snx11029n017(ib0:1)    #oss13
10.10.101.19   snx11029n018(ib0:1)    #oss14
10.10.101.20   snx11029n019(ib0:1)    #oss15
10.10.101.21   snx11029n020(ib0:1)    #oss16
10.10.101.22   snx11029n021(ib0:1)    #oss17
10.10.101.23   snx11029n022(ib0:1)    #oss18
10.10.101.24   snx11029n023(ib0:1)    #oss19
10.10.101.25   snx11029n024(ib0:1)    #oss20
10.10.101.26   snx11029n025(ib0:1)    #oss21
10.10.101.27   snx11029n026(ib0:1)    #oss22
10.10.101.28   snx11029n027(ib0:1)    #oss23
```

Move the `cluster-name.ib` file to the working directory on the SMW or boot node where the `clcvt` command will be run.

10.3.6 Create the CLCVT client-system.rtrIm File

About this task

The `client-system.rtrIm` file contains output from the `rtr -Im` command as executed from the SMW. When capturing the command output to a file, use the `-H` option to remove the header information from `rtr -Im` or open the file after capturing and delete the first two lines.

Procedure

1. Log on to the SMW as `crayadm`.

2. Run the `rtr -Im` command and capture the output (without header information) to a file.

```
crayadm@smw> rtr -Im -H > client-system.rtrIm
```

Example *client-system*.rtrIm File

```
  4        4       c0-0c2s2n1    c0-0c2s2g0     0  0  0
  5        5       c0-0c2s2n2    c0-0c2s2g0     0  0  0
221      221       c1-0c0s7n1    c1-0c0s7g0     0  3  7
222      222       c1-0c0s7n2    c1-0c0s7g0     0  3  7
641      769       c3-0c1s0n1    c3-0c1s0g0     1  4  0
642      770       c3-0c1s0n2    c3-0c1s0g0     1  4  0
661      789       c3-0c1s5n1    c3-0c1s5g0     1  4  5
662      790       c3-0c1s5n2    c3-0c1s5g0     1  4  5
```

```
721       849        c3-0c2s4n1    c3-0c2s4g0     1  5  4
722       850        c3-0c2s4n2    c3-0c2s4g0     1  5  4
797       1053       c4-0c0s7n1    c4-0c0s7go     2  0  7
798       1054       c4-0c0s7n2    c4-0c0s7go     2  0  7
1025      1281       c5-0c1s0n1    c5-0c1s0g0     2  4  0
1046      1302       c5-0c1s0n2    c5-0c1s0g0     2  4  0
1045      1301       c5-0c1s5n1    c5-0c1s5g0     2  4  5
1046      1302       c5-0c1s5n2    c5-0c1s5g0     2  4  5
1077      1365       c5-0c2s4n1    c5-0c2s4g0     2  4  13
1078      1366       c5-0c2s4n2    c5-0c2s4g0     2  4  13
```

3. Move the *client-system*.rtrIm (hera.rtrIm) file to the working directory where the clcvt command will be run.

10.3.7 Generate ip2nets and routes Information

After the prerequisite files have been created, generate the persistent-storage file with the clcvt generate action. This portable file will then be used to create ip2nets and routes directives for the servers, routers, and clients.

The following procedures frequently use the --split-routes=4 flag, which prints information that can be loaded into ip2nets and routes files. This method of adding modprobe.conf directives is particularly valuable for large systems where the directives might otherwise exceed the modprobe buffer limit.

- *Create the CLCVT persistent-storage File* on page 75
- *Create ip2nets and Routes Information for Compute Nodes* on page 76
- *Create ip2nets and routes Information for Service Node Lustre Clients (MOM and Internal Login Nodes)* on page 76
- *Create ip2nets and routes Information for LNet Router Nodes* on page 77
- *Create ip2nets and routes Information for Lustre Server Nodes* on page 78

10.3.7.1 Create the CLCVT persistent-storage File

Procedure

1. Move all prerequisite files to an empty directory on the boot node or SMW (the clcvt command is only available on the boot node or the SMW).

   The working directory should include the following files.

```
crayadm@smw$ ll
total 240
-rw-rw-r-- 1 crayadm crayadm  23707 Oct  8 14:27 hera.hosts
-rw-rw-r-- 1 crayadm crayadm    548 Oct  8 14:27 hera.ib
-rw-rw-r-- 1 crayadm crayadm  36960 Oct  8 14:27 hera.rtrIm
-rw-rw-r-- 1 crayadm crayadm   1077 Feb  8 14:27 info.snx11029n
-rw-rw-r-- 1 crayadm crayadm    662 Feb  8 14:27 snx11029n.ib
```

2. Create the persistent-storage file. (Use the --debug flag to display debugging information.)

```
crayadm@smw$ clcvt generate
INFO:LNET_filesystem.load: clustername = snx11029n
INFO:LNET_filesystem.load: This is a SONEXION
INFO:LNET_filesystem.load: 12 SSU, 6 SSU per rack
INFO:LNET_filesystem.load: You have chosen to use Fine Grained Routing.
INFO:LNET_filesystem.load: The ratio of LNET router connections to OSSs is '4:3'.
```

10.3.7.2 Create ip2nets and Routes Information for Compute Nodes

Procedure

1. Execute the `clcvt` command with the `compute` flag to generate directives for the compute nodes.

```
crayadm@smw$ clcvt compute --split-routes=4
# Place the following line(s) in the appropriate 'modprobe' file.
#vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
options lnet ip2nets=/path/to/ip2nets-loading/filename
options lnet routes=/path/to/route-loading/filename
#^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
# Place the following line(s) in the appropriate ip2nets-loading file.
#vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
gni1 10.128.*.*
#^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
# Place the following line(s) in the appropriate route-loading file.
#vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
o2ib6000 1 [4,5]@gni1
o2ib6002 1 [222,642,662,722]@gni1
o2ib6003 1 [798,1026,1046,1078]@gni1
o2ib6004 1 [221,641,661,721]@gni1
o2ib6005 1 [797,1025,1045,1077]@gni1
o2ib6006 1 [222,642,662,722]@gni1
o2ib6007 1 [798,1026,1046,1078]@gni1
o2ib6008 1 [221,641,661,721]@gni1
o2ib6009 1 [797,1025,1045,1077]@gni1
o2ib6000 2 [221,222,641,642,661,662,721,722,797,798,1025,1026,1045,1046,1077,1078]@gni1
o2ib6002 2 [221,641,661,721]@gni1
o2ib6003 2 [797,1025,1045,1077]@gni1
o2ib6004 2 [222,642,662,722]@gni1
o2ib6005 2 [798,1026,1046,1078]@gni1
o2ib6006 2 [221,641,661,721]@gni1
o2ib6007 2 [797,1025,1045,1077]@gni1
o2ib6008 2 [222,642,662,722]@gni1
o2ib6009 2 [798,1026,1046,1078]@gni1
#^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
```

2. Place the command output in the appropriate `modprobe` configuration file.

10.3.7.3 Create ip2nets and routes Information for Service Node Lustre Clients (MOM and Internal Login Nodes)

Procedure

1. Execute the `clcvt` command with the `login` flag to generate directives for the service node Lustre clients.

```
crayadm@smw$  clcvt login --split-routes=4
# Place the following line(s) in the appropriate 'modprobe' file.
#vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
options lnet ip2nets=/path/to/ip2nets-loading/filename
options lnet routes=/path/to/route-loading/filename
#^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
# Place the following line(s) in the appropriate ip2nets-loading file.
#vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
gni1 10.128.*.*
#^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
# Place the following line(s) in the appropriate route-loading file.
#vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
o2ib6000 1 [4,5]@gni1
o2ib6002 1 [222,642,662,722]@gni1
o2ib6003 1 [798,1026,1046,1078]@gni1
o2ib6004 1 [221,641,661,721]@gni1
o2ib6005 1 [797,1025,1045,1077]@gni1
```

```
o2ib6006 1 [222,642,662,722]@gni1
o2ib6007 1 [798,1026,1046,1078]@gni1
o2ib6008 1 [221,641,661,721]@gni1
o2ib6009 1 [797,1025,1045,1077]@gni1
o2ib6000 2 [221,222,641,642,661,662,721,722,797,798,1025,1026,1045,1046,1077,1078]@gni1
o2ib6002 2 [221,641,661,721]@gni1
o2ib6003 2 [797,1025,1045,1077]@gni1
o2ib6004 2 [222,642,662,722]@gni1
o2ib6005 2 [798,1026,1046,1078]@gni1
o2ib6006 2 [221,641,661,721]@gni1
o2ib6007 2 [797,1025,1045,1077]@gni1
o2ib6008 2 [222,642,662,722]@gni1
o2ib6009 2 [798,1026,1046,1078]@gni1
#^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
```

2. Place the command output in the appropriate `modprobe` configuration file.

10.3.7.4 Create ip2nets and routes Information for LNet Router Nodes

Procedure

1. Execute the `clcvt` command with the `router` flag to generate directives for the LNet router nodes.

```
crayadm@smw$  clcvt router --split-routes=4
# Place the following line(s) in the appropriate 'modprobe' file.
#vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
options lnet ip2nets=/path/to/ip2nets-loading/filename
options lnet routes=/path/to/route-loading/filename
#^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
# Place the following line(s) in the appropriate ip2nets-loading file.
#vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
gni1 10.128.*.*
o2ib6000 10.10.100.[101,105,109,113,117,121,125,129,133]
o2ib6000 10.10.101.[104,108,112,116,120,124,128,132]
o2ib6000(ib2) 10.10.100.102
o2ib6002 10.10.100.[105,109,113,117]
o2ib6002 10.10.101.[104,108,112,116]
o2ib6003 10.10.100.[121,125,129,133]
o2ib6003 10.10.101.[120,124,128,132]
o2ib6004 10.10.100.[105,109,113,117]
o2ib6004 10.10.101.[104,108,112,116]
o2ib6005 10.10.100.[121,125,129,133]
o2ib6005 10.10.101.[120,124,128,132]
o2ib6006(ib2) 10.10.100.[103,107,111,115]
o2ib6006(ib2) 10.10.101.[106,110,114,118]
o2ib6007(ib2) 10.10.100.[119,123,127,131]
o2ib6007(ib2) 10.10.101.[122,126,130,134]
o2ib6008(ib2) 10.10.100.[103,107,111,115]
o2ib6008(ib2) 10.10.101.[106,110,114,118]
o2ib6009(ib2) 10.10.100.[119,123,127,131]
o2ib6009(ib2) 10.10.101.[122,126,130,134]
#^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
# Place the following line(s) in the appropriate route-loading file.
#vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
o2ib6000 1 [4,5]@gni1
o2ib6002 1 [222,642,662,722]@gni1
o2ib6003 1 [798,1026,1046,1078]@gni1
o2ib6004 1 [221,641,661,721]@gni1
o2ib6005 1 [797,1025,1045,1077]@gni1
o2ib6006 1 [222,642,662,722]@gni1
o2ib6007 1 [798,1026,1046,1078]@gni1
o2ib6008 1 [221,641,661,721]@gni1
o2ib6009 1 [797,1025,1045,1077]@gni1
o2ib6000 2 [221,222,641,642,661,662,721,722,797,798,1025,1026,1045,1046,1077,1078]@gni1
o2ib6002 2 [221,641,661,721]@gni1
o2ib6003 2 [797,1025,1045,1077]@gni1
o2ib6004 2 [222,642,662,722]@gni1
o2ib6005 2 [798,1026,1046,1078]@gni1
o2ib6006 2 [221,641,661,721]@gni1
o2ib6007 2 [797,1025,1045,1077]@gni1
```

```
o2ib6008 2 [222,642,662,722]@gni1
o2ib6009 2 [798,1026,1046,1078]@gni1
#^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
```

2. Place the command output in the appropriate `modprobe` configuration file.

10.3.7.5 Create ip2nets and routes Information for Lustre Server Nodes

Procedure

1. Execute the `clcvt` command with the `server` flag to generate directives for the Lustre server nodes.

```
crayadm@smw$ clcvt server --split-routes=4
# Place the following line(s) in the appropriate 'modprobe' file.
#vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
options lnet ip2nets=/path/to/ip2nets-loading/filename
options lnet routes=/path/to/route-loading/filename
#^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
# Place the following line(s) in the appropriate ip2nets-loading file.
#vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
o2ib6 10.10.*.*
o2ib6000 10.10.100.[3,4]
o2ib6002 10.10.100.[5,7,9]
o2ib6003 10.10.100.[6,8,10]
o2ib6004 10.10.100.[11,13,15]
o2ib6005 10.10.100.[12,14,16]
o2ib6006(ib0:1) 10.10.101.[17,19,21]
o2ib6007(ib0:1) 10.10.101.[18,20,22]
o2ib6008(ib0:1) 10.10.101.[23,25,27]
o2ib6009(ib0:1) 10.10.101.[24,26,28]
#^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
# Place the following line(s) in the appropriate route-loading file.
#vvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvvv
gni1 1 10.10.100.[101,102]@o2ib6000
gni1 1 10.10.100.[105,109,113,117]@o2ib6002
gni1 1 10.10.100.[121,125,129,133]@o2ib6003
gni1 1 10.10.101.[104,108,112,116]@o2ib6004
gni1 1 10.10.101.[120,124,128,132]@o2ib6005
gni1 1 10.10.101.[106,110,114,118]@o2ib6006
gni1 1 10.10.101.[122,126,130,134]@o2ib6007
gni1 1 10.10.100.[103,107,111,115]@o2ib6008
gni1 1 10.10.100.[119,123,127,131]@o2ib6009
gni1 2 10.10.100.[105,109,113,117,121,125,129,133]@o2ib6000
gni1 2 10.10.101.[104,108,112,116,120,124,128,132]@o2ib6000
gni1 2 10.10.101.[104,108,112,116]@o2ib6002
gni1 2 10.10.101.[120,124,128,132]@o2ib6003
gni1 2 10.10.100.[105,109,113,117]@o2ib6004
gni1 2 10.10.100.[121,125,129,133]@o2ib6005
gni1 2 10.10.100.[103,107,111,115]@o2ib6006
gni1 2 10.10.100.[119,123,127,131]@o2ib6007
gni1 2 10.10.101.[106,110,114,118]@o2ib6008
gni1 2 10.10.101.[122,126,130,134]@o2ib6009
#^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
```

2. Place the command output in the appropriate `modprobe` configuration file for the Lustre servers. (For more information, refer to the site-specific Lustre server documentation.)

10.4 Place the LNet Configuration on a CLE6 System

10.4.1 Copy Files for External Lustre Fine-grained Routing

Prerequisites

This procedure is only for systems that use an external Lustre file system. It assumes the following:

- Fine-grained routing (FGR) files have been generated by `clcvt`
- Cray LNet configuration service (`cray_lnet`) has been configured with FGR

About this task

This procedure places the `ip2nets.conf` and `routes.conf` files in the CLE config set for the LNet routers.

Procedure

1. Create an `lnet` directory under `roles` in the CLE config set directory structure.

   This example uses a config set named *p0*. Substitute the correct config set name for this site.

   ```
   smw# mkdir -p /var/opt/cray/imps/config/sets/p0/files/roles/lnet
   ```

2. Confirm the file names of the fine-grained routing files.

   It is possible that these two files were created with names other than `ip2nets.conf` and `routes.conf`. Check these two settings in the `cray_lnet` configuration service to see what file names are used (example settings are for config set *p0* and a file system with key *sonexion*).

   ```
   smw# cfgset search -l advanced -s cray_lnet -t fgr_routes p0
   # 2 matches for 'fgr_routes' from cray_lnet_config.yaml
   #--------------------------------------------------------------
   cray_lnet.settings.fgr_routes.data.sonexion.ip2nets_file: ip2nets.conf
   cray_lnet.settings.fgr_routes.data.sonexion.routes_file: routes.conf
   ```

3. Copy the `ip2nets.conf` and `routes.conf` files to the `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
   ```

10.4.2 Configure LNet Routers for Fine-Grained Routing

Prerequisites

Fine-grained routing files have been configured using `clcvt`.

About this task

This procedure modifies the *Cray System Management Configuration Worksheet* for the `cray_lnet` service in the config set to configure fine-grained routing (FGR).

Procedure

1. As `root`, use `cfgset` to modify the `cray_lnet` service in the configuration set.

```
smw# cfgset update --service cray_lnet --mode interactive partition
  cray_lnet        [ status: enabled ]  [ validation: valid ]

---------------------------------------------------------------------------
  Selected     #      Settings              Value/Status (level=basic)
---------------------------------------------------------------------------
                      ko2iblnd
               1)       peer_credits          63
               2)       concurrent_sends      63
                      local_lnet
               3)       lnet_name             gni1
               4)       ip_wildcard           10.128.*.*
               5)     flat_routes             [ 6 sub-settings unconfigured, select
                                              and enter C to add entries ]
               6)     fgr_routes              [ 5 sub-settings unconfigured, select
                                              and enter C to add entries ]
---------------------------------------------------------------------------
```

2. Enter `6`, then `C` to configure fine-grained routing (`fgr_routes`).

3. Enter + and type the Sonexion server designation (`snx11029n`) to configure `cray_lnet.settings.fgr_routes`.

```
cray_lnet.settings.fgr_routes.data.dest_name
[<cr>=set '', <new value>, ?=help, @=less] $ snx11029n
CUG
```

4. Enter + to configure `cray_lnet.settings.fgr_routes.data.snx11029n.routers`.

```
cray_lnet.settings.fgr_routes.data.snx11029n.routers
[<cr>=set 0 entries, +=add an entry, ?=help, @=less] $ +
Add routers (Ctrl-d to exit) $ c0-0c2s2n1
Add routers (Ctrl-d to exit) $ c0-0c2s2n2
Add routers (Ctrl-d to exit) $ c1-0c0s7n1
Add routers (Ctrl-d to exit) $ c1-0c0s7n2
Add routers (Ctrl-d to exit) $ c3-0c1s0n1
Add routers (Ctrl-d to exit) $ c3-0c1s0n2
...
Add routers (Ctrl-d to exit) $
```

5. Specify the name of the `ip2nets.conf` file for this FGR configuration. (The file must be placed in the config set on the SMW in `/var/opt/cray/imps/config/sets/`*`config_set`*`/files/roles/lnet/`.)

```
cray_lnet.settings.fgr_routes.data.snx11029n.ip2nets_file
[<cr>=set '', <new value>, ?=help, @=less] $ ip2nets.conf
```

6. Specify the name of the `routes.conf` file for this FGR configuration. Place `routes.conf` in the `config_set` on the SMW `/var/opt/cray/imps/config/sets/`*`config_set`*`/files/roles/lnet/`.

```
cray_lnet.settings.fgr_routes.data.snx11029n.routes_file
[<cr>=set '', <new value>, ?=help, @=less] $ routes.conf
```

7. Follow the guidance in the `cray_lnet` worksheet and set each configuration value for the system as needed.

8. Review the configuration settings:

```
Configured Values:
1) 'snx11029n'
    a) routers:
        c0-0c2s2n1
        c0-0c2s2n2
        c1-0c0s7n1
        c1-0c0s7n2
        ...
    b) ip2nets_file: ip2nets.conf
    c) routes_file: routes.conf
    d) ko2iblnd_peer_credits: 63
    e) ko2iblnd_concurrent_sends: 63
```

9. Reboot the nodes associated with these configuration changes and integrate the new configuration into the default config set (`p0`).

Configure Sonexion system with this FGR scheme.

10.5 Create Custom LNet Configuration for Sonexion

Prerequisites

For a new system, first complete the setup procedures described in the *Sonexion Field Installation Guide*.

About this task

Follow this procedure to create a custom LNet configuration on the Sonexion system while in the "daily mode".

Procedure

1. Log in to the primary management mode.

2. Change to `root`.

   ```
   $ sudo su -
   ```

3. Stop the Lustre file system.

   ```
   # cscli unmount -f file_system_name
   ```

4. Use the following steps to change the `o2ib` index. First, start the MySQL client and connect to the `t0db` database.

   ```
   # mysql t0db
   ```

5. Display the `mgsNID`, `nidFormat`, and `nidIndex` entries.

   ```
   mysql> select * from property where name in ('nidFormat', 'mgsNID', 'nidIndex');
   +-----+--------------------+-----------+---------------------+-----------+
   | id  | context            | name      | value               | attr_type |
   +-----+--------------------+-----------+---------------------+-----------+
   |  22 | snx11000n:beConfig | nidFormat | l%s@o2ib%d          | str       |
   | 106 | snx11000n:beConfig | nidIndex  | 3                   | int       |
   | 109 | snx11000n:beConfig | mgsNID    | lsnx11000n002@o2ib0 | str       |
   +-----+--------------------+-----------+---------------------+-----------+
          3 rows in set (0.00 sec)
   ```

6. Set the `o2ib` index by modifying the `nidIndex` entry.

   ```
   mysql> update property set value=desired_odib_index where name='nidIndex';
   ```

   For example:

   ```
   mysql> update property set value=2 where name='nidIndex';
   Query OK, 1 row affected (0.02 sec)
   Rows matched: 1  Changed: 1  Warnings: 0
   ```

7. Set the `mgsNID` entry to match the `o2ib` index.

   ```
   mysql> update property set \
   value='original_value@o2ibdesired_o2ib_index' \
   where name='mgsNID';
   ```

   For example:

   ```
   mysql> update property set value='lsnx11029n002@o2ib0' where name='mgsNID';
   Query OK, 1 row affected (0.04 sec)
   Rows matched: 1  Changed: 1  Warnings: 0
   ```

8. Verify the changes.

   ```
   mysql> select * from property where name in ('nidFormat', 'mgsNID', 'nidIndex');
   +-----+--------------------+-----------+---------------------+-----------+
   | id  | context            | name      | value               | attr_type |
   +-----+--------------------+-----------+---------------------+-----------+
   |  22 | snx11000n:beConfig | nidFormat | l%s@o2ib%d          | str       |
   | 106 | snx11000n:beConfig | nidIndex  | 2                   | int       |
   | 109 | snx11000n:beConfig | mgsNID    | lsnx11000n002@o2ib0 | str       |
   +-----+--------------------+-----------+---------------------+-----------+
          3 rows in set (0.00 sec)
   ```

9. Close the MySQL session.

   ```
   mysql> quit
   ```

10. Run `puppet`.

    ```
    # /opt/xyratex/bin/beUpdatePuppet -sa
    ```

11. Run the `beSystemNetConfig.sh` script on the primary management node and wait for it to finish.

    ```
    # /opt/xyratex/bin/beSystemNetConfig.sh \
    -c file_location/lnet.conf -r file_location/routes.conf \
    -i file_location/ip2nets.conf clustername
    ```

12. Verify that the customized LNet configuration has been applied.

    a. List the `nids`.

       ```
       # pdsh -g lustre lctl list_nids | sort
       ```

    b. List the nodes and targets.

       ```
       # cscli fs_info
       ```

13. Start the Lustre file system and wait for the targets to mount on all system nodes.

    ```
    # cscli mount -f file_system_name
    ```

14. Modify `modprobe.conf` to support IB Aliases.

    The following examples intercept the LNet module and setup IB aliases before the module is loaded. **Each example below is a single line** in `lnet.conf`.

    a. Add `ib0:1` to `10.10.101` subnet.

       ```
       ## Create aliases.
       ## Intercepts the Lnet module and sets up the aliases before the module is loaded install lnet

       /sbin/ip -o -4 a show ib0 | /usr/bin/awk '/inet/{s=$4;sub("10\\.10\\.100\
       \.","10.10.101.",s);print "/sbin/ip address add dev ib0 label ib0:1",s}' | /bin/sh ;/sbin/
       modprobe --ignore-install lnet $CMDLINE_OPTS
       ```

b. Remove all `ib0:1` aliases when LNet module is unloaded.

```
## Remove all ib0:1 aliases when lnet module is unloaded

remove lnet /sbin/modprobe -r --ignore-remove lnet && /sbin/ip -o -4 a show label ib0:1 | awk
'{print "/sbin/ip address del dev ib0 label ib0:1",$4}' | /bin/sh
```

c. Configure a metadata server on a 3rd IPv4 subnet.

```
install lnet if nodeattr mds; then /sbin/ip -o -4 a show ib0 | /usr/bin/awk '/inet/{s=$4;sub("10\
\.10\\.100\\.","10.10.102.",s);print "/sbin/ip address add dev ib0 label ib0:1",s}' | /bin/sh ;
else /sbin/ip -o -4 a show ib0 | /usr/bin/awk '/inet/{s=$4;sub("10\\.10\\.100\
\.","10.10.101.",s);print "/sbin/ip address add dev ib0 label ib0:1",s}' | /bin/sh ; fi ; /sbin/
modprobe --ignore-install lnet $CMDLINE_OPTS
```

11 Perform the First-Run Configuration

About this task

After the Sonexion Cray Sonexion System Manager system is installed and powered on, launch the CSM wizard and perform the first-run configuration procedures. After all the procedures are complete, the system is fully configured for the customer environment and ready for use.

> **IMPORTANT:** Throughout the first-run configuration procedure, the **Next** button must be clicked on a number of screens. Do not click **Next** too quickly from one screen to the next, or the configuration procedure may fail.

Procedure

1. To launch CSM wizard, open a Java-enabled browser (Google Chrome preferred).
2. Enter the CSM wizard IP address in the browser's URL field.

   The CSM wizard launches and provides prompts to perform the first-run procedures, beginning with entering the default login and password.
3. Enter the factory-set user name `admin` and login `admin`.

   The factory login is `admin` and default password is `admin`. This should be changed as part of the first run configuration procedures.

11.1 Create the Administrator Password

Prerequisites

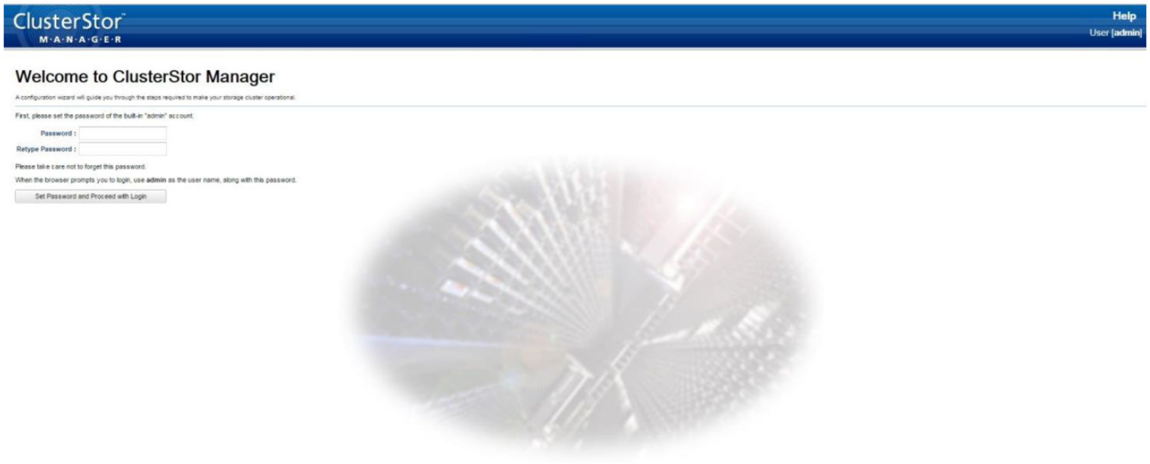
System is on and the CSM wizard is launched.

If the Site Prep questionnaire was filled out as part of the purchase, much of the necessary information in this procedure will be pre-populated.

About this task

When CSM launches for the first time, the **Welcome** screen appears, and prompts to create a new `admin` password. This prompt no longer appears after the administrator password is set.

Figure 18. CCM Welcome Screen



Welcome to ClusterStor Manager

A configuration wizard will guide you through the steps required to make your storage cluster operational.

First, please set the password of the built-in "admin" account.

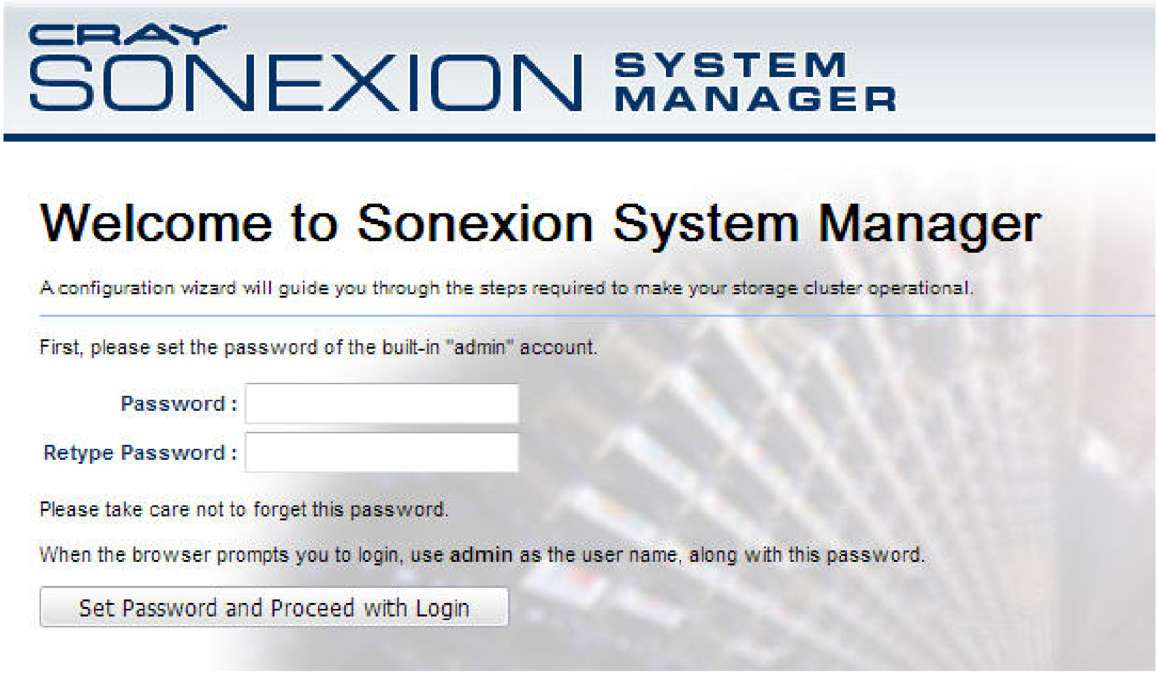
Password :

Retype Password :

Please take care not to forget this password.

When the browser prompts you to login, use **admin** as the user name, along with this password.

Figure 19. CSM Welcome Screen



Welcome to Sonexion System Manager

A configuration wizard will guide you through the steps required to make your storage cluster operational.

First, please set the password of the built-in "admin" account.

Password :

Retype Password :

Please take care not to forget this password.

When the browser prompts you to login, use **admin** as the user name, along with this password.

Procedure

1. Create the administrator password.

   The password is limited to 8 characters.

2. Re-enter the new administrator password to confirm.

3. Click **Set Password and Proceed with Login**.

4. Monitor progress:
   - If the **License** screen appears, proceed to Acknowledge License Screen.
   - **Do not** click **OK** if an error window opens. Follow the procedure below:

     Troubleshoot the CSM Welcome Screen

   a. To resolve **"node has not received changes"** error, which can occur after selecting the **Set Password and Proceed with Login** button, do not click the **OK** box on the error.

   b. Carefully inspect any errors generated by CSM. Often, the errors reveal the exact problem through backend script output, middleware issues, etc.

      This error usually points to larger underlying issues, the most common of which is that one or more node(s) is powered off. If this is the case:

      1. Determine if there are downed nodes:

         ```
         [MGMT0]$ pm -q
         ```

      2. Power the node(s) on:

         ```
         [MGMT0]$ pm -1 nodename
         ```

      3. Verify that the nodes power on correctly. Log in to the nodes when power on completes:

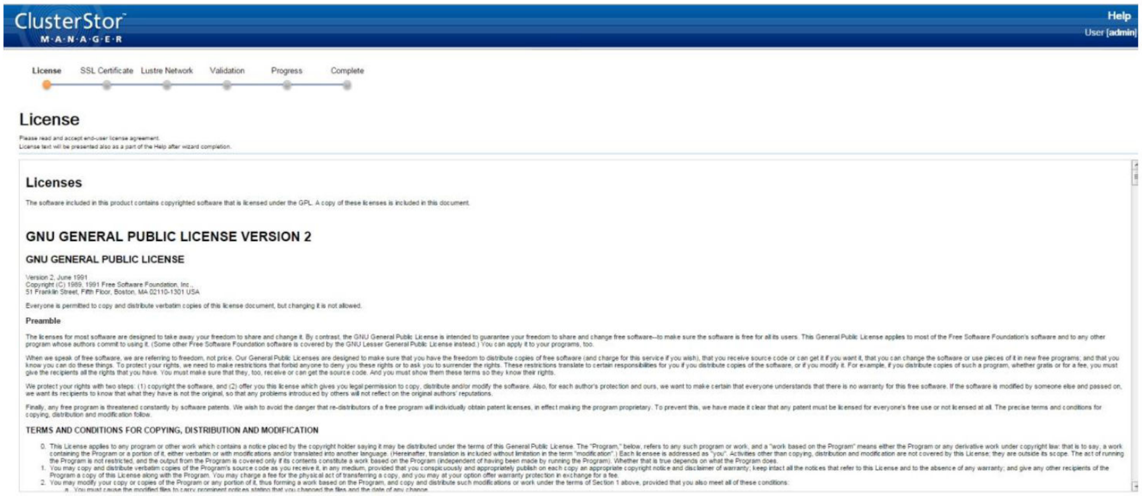
         ```
         [MGMT0]$ ssh nodename
         ```

      4. Repeat the preceding steps to create the administrator password.

5. Click **OK**.

   When the password is accepted, the **License** screen appears.

Figure 20. License Screen



License

Please read and accept end-user license agreement.
License text will be presented also as a part of the Help after wizard completion.

Licenses

The software included in this product contains copyrighted software that is licensed under the GPL. A copy of these licenses is included in this document.

GNU GENERAL PUBLIC LICENSE VERSION 2

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991
Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things. To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it. For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

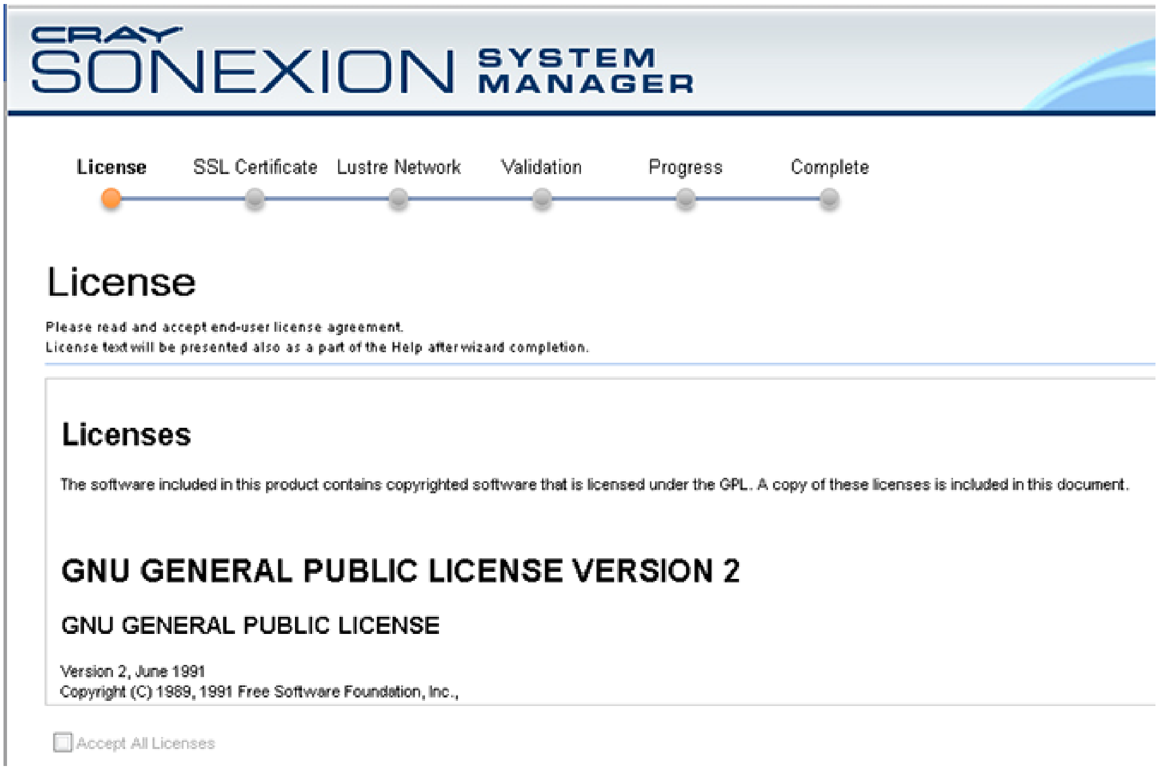
We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software. Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that re-distributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all. The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program," below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you". Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.
1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program. You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.
2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:
   a. You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

Figure 21. License Screen



License

Please read and accept end-user license agreement.
License text will be presented also as a part of the Help after wizard completion.

Licenses

The software included in this product contains copyrighted software that is licensed under the GPL. A copy of these licenses is included in this document.

GNU GENERAL PUBLIC LICENSE VERSION 2

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991
Copyright (C) 1989, 1991 Free Software Foundation, Inc.,

□ Accept All Licenses

6. Read the license.

7. Scroll through the list of licenses and select the **Accept All Licenses** box.

   The **Accept All Licenses** box remains greyed out until the list of licenses has been scrolled through.

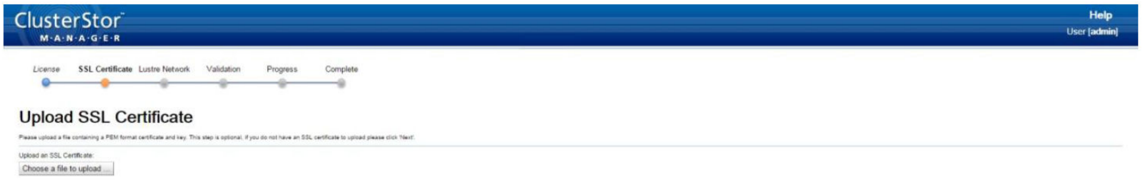
8. Click **Next**.

   The **Upload SSL Certificate** screen displays (optional).

   Sonexion 3000 and 3000t systems support only privacy enhanced mail (`.PEM`) format files. If the SSL certificate file is in a different format, convert it to `.PEM` before continuing with this procedure.

   > **TIP:** Uploading an SSL certificate is an optional step. To bypass this procedure and proceed to configure the Lustre network interfaces, click **Next** now.

Figure 22. SSL Certificate Screen

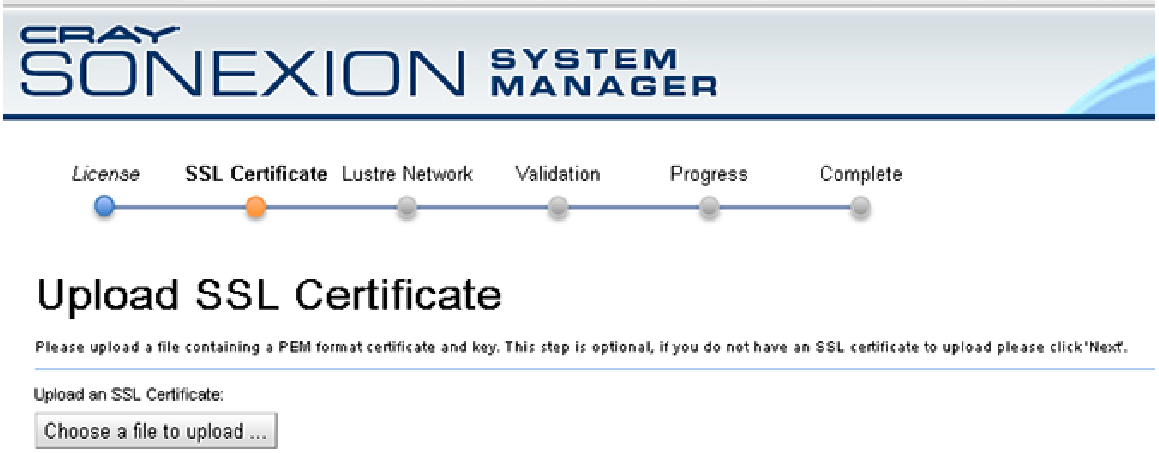


Upload SSL Certificate

Please upload a file containing a PEM format certificate and key. This step is optional. If you do not have an SSL certificate to upload please click 'Next'.

Upload an SSL Certificate:

Figure 23. SSL Certificate Screen



Upload SSL Certificate

Please upload a file containing a PEM format certificate and key. This step is optional, if you do not have an SSL certificate to upload please click 'Next'.

Upload an SSL Certificate:

9. When the **Upload SSL Certificate** screen appears, click **Choose a file to upload...**.

10. Locate and select the SSL certificate (in `.PEM` format).

11. Click **Open**.

12. When a green checkmark appears next to the filename, click **Next**.

    The SSL certificate uploads and a content box displays information about the certificate.

    > **IMPORTANT:** Sonexion 3000 and 3000t support LDAP over SSL (LDAPS), which requires an SSL certificate. The certificate uploaded at this step is for general secure access to the management console and is different from the SSL certificate required for LDAPS. For more information, refer to *Specify Lustre User Authentication via the LDAP Screen*.

    The **Lustre Network** screen displays. Proceed to *Configure the Lustre Network Interfaces* on page 88.

11.2 Configure the Lustre Network Interfaces

Prerequisites

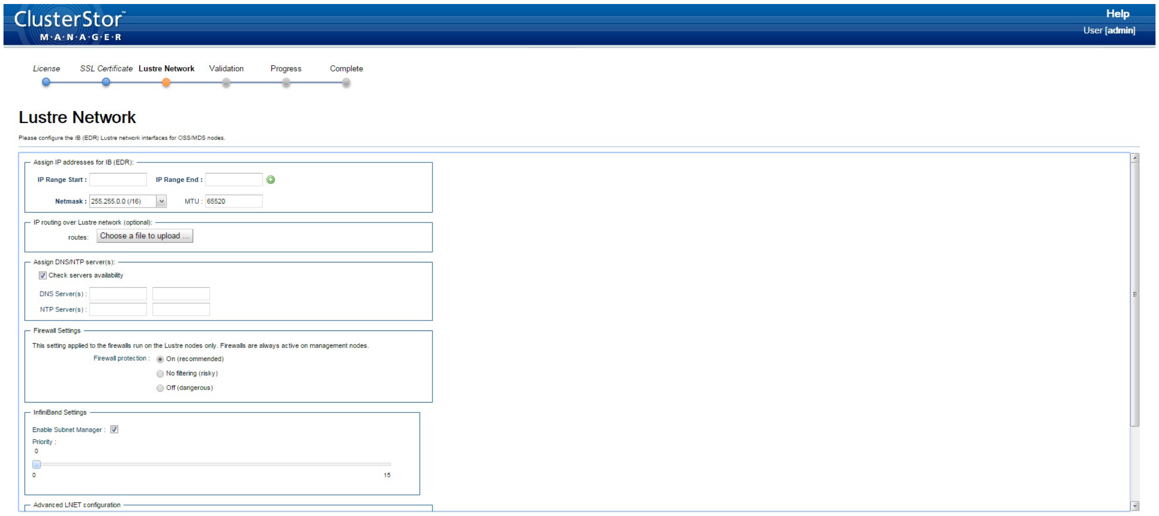
The system is powered on, the administrator password has been created, the license has been accepted, and the SSL certificate has been uploaded, if desired. The **Lustre Network** screen has appeared.

About this task

This procedure configures the Lustre network interfaces by specifying IP addresses for the nodes and DNS and NTP servers, along with the netmask address and the maximum transmit unit (MTU) value.

Sonexion 3000 and 3000t systems support InfiniBand EDR, Omni-Path, 100GbE, or 40GbE networking on the local data network (LDN).

Figure 24. Lustre Network Screen



Lustre Network

Please configure the IB (EDR) Lustre network interfaces for OSS/MDS nodes.

Assign IP addresses for IB (EDR):

IP Range Start : **IP Range End :**

Netmask : 255.255.0.0 (/16) MTU : 65520

IP routing over Lustre network (optional):

routes: Choose a file to upload ...

Assign DNS/NTP server(s):

☑ Check servers availability

DNS Server(s) :

NTP Server(s) :

Firewall Settings

This setting applied to the firewalls run on the Lustre nodes only. Firewalls are always active on management nodes.

Firewall protection : ● On (recommended)
○ No filtering (risky)
○ Off (dangerous)

InfiniBand Settings

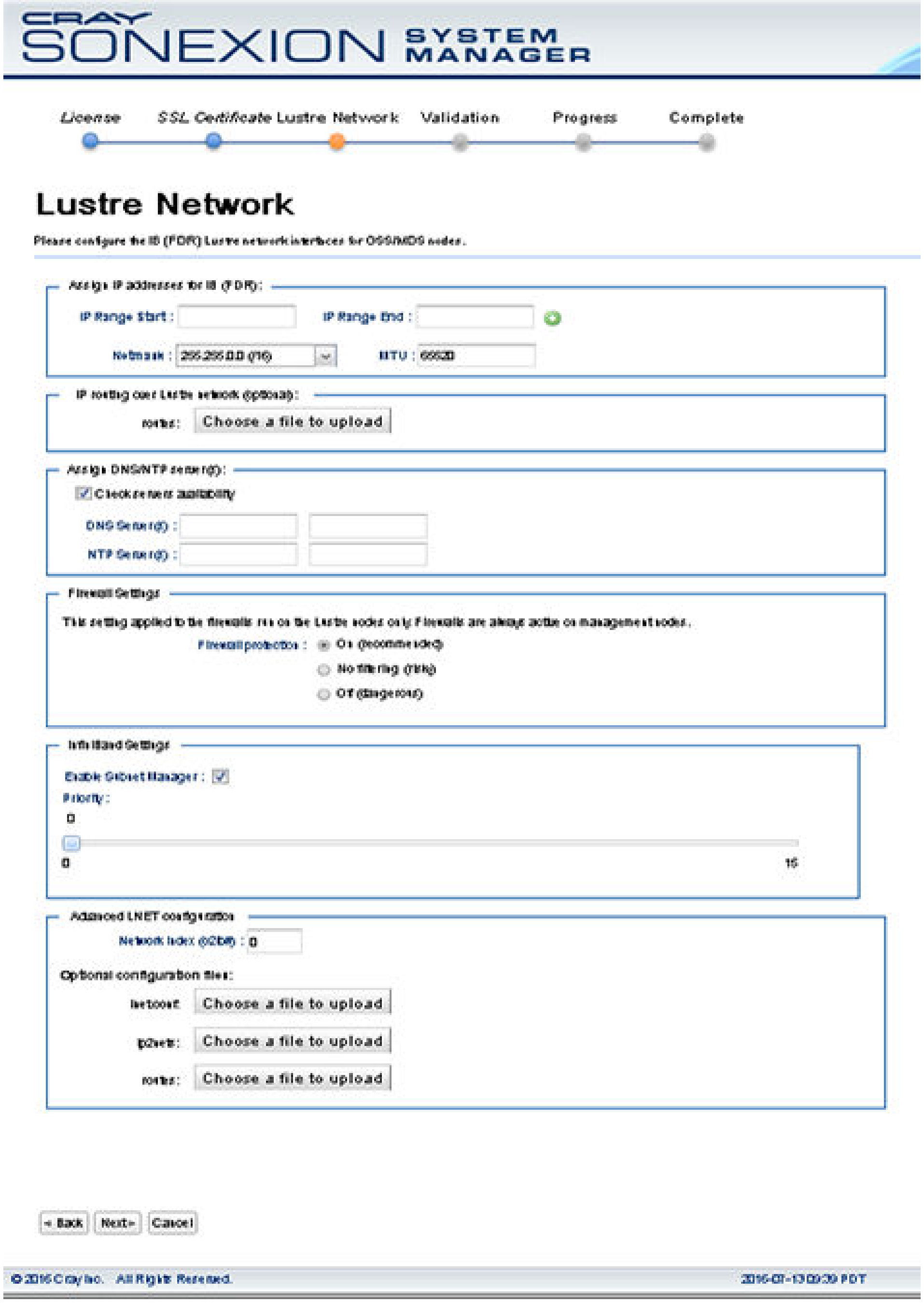
Enable Subnet Manager : ☑

Priority :
0

0 15

Advanced LNET configuration

Figure 25. Lustre Network Screen



Lustre Network

Please configure the IB (FDR) Lustre network interfaces for OSS/MDS nodes.

Assign IP addresses for IB (FDR):

IP Range Start : 　 IP Range End :

Netmask : 255.255.0.0 (/16) 　 MTU : 65520

IP routing over Lustre network (optional):

routes: Choose a file to upload

Assign DNS/NTP server(s):

☑ Check servers availability

DNS Server(s) :

NTP Server(s) :

Firewall Settings

This setting applies to the firewalls rules on the Lustre nodes only. Firewalls are always active on management nodes.

Firewall protection : ● On (recommended)
○ No filtering (risky)
○ Off (dangerous)

InfiniBand Settings

Enable Subnet Manager : ☑

Priority :
0

0 　 15

Advanced LNET configuration

Network index (o2ib#) : 0

Optional configuration files:

lnet.conf: Choose a file to upload

ip2nets: Choose a file to upload

routes: Choose a file to upload

© 2016 Cray Inc. All Rights Reserved. 　 2016-07-13 09:39 PDT

Procedure

1. Specify IP addresses for nodes (n000-n*xxx*) as a single range or as multiple ranges.

   It is the administrator's choice whether to specify a single IP address range for all Lustre nodes in the system or use multiple IP address ranges for different nodes (for example, one IP address range for the MGS and MDS nodes, and a separate IP address range for the OSS nodes). To specify multiple IP address ranges, enter one IP address range and click **Plus/+** to enter additional IP address ranges.

In the **Assign IP Addresses for IB/40GbE** group, enter the following:

a. **IP Range Start** address

b. **IP Range End** address for range number of nodes +1

If an incorrect IP address is entered, the field entry changes to red, indicating a problem. Enter a valid IP address and continue.

2. Enter the **Netmask** address. To select from a list of standard netmask addresses, click the **down** arrow.

3. Enter the **MTU** value (in bytes).

a. If InfiniBand is used as the high-speed network, set the default MTU value to 65520, which supports jumbo frames (JF) traffic. This is the optimal setting for IP over InfiniBand (IPoIB).

b. If 40GbE is used as the high-speed network, Cray recommends setting the MTU value to 9000 to enable support of jumbo frames over Ethernet.

c. Use default MTU settings for OPA. QUESTION: Is this what we recommend?

d. Contact Cray Service Support for 100GbE MTU settings. QUESTION: Has 100Gb/s been determined?

> **IMPORTANT:** Be sure all machines and switches on the network are configured to the same MTU value.

4. Enter the **DNS Server(s)** IP addresses, if not already discovered.

5. Enter the **NTP Server(s)** IP addresses.

This screen provides the ability to change optional parameters, firewall settings, IB subnet manager, and network index, and upload optional LNet configuration files. A custom LNet configuration can be set up after the installation if LNet config files are not prepared. Contact Cray Support for clarification.

6. Optional: to verify the DNS and NTP IP addresses entered earlier, check the appropriate box.

7. Click **Next**.

The **Lustre Network in Progress** window remains open while the network settings are applied. After the network settings are applied, the **In Progress** window closes automatically and the **Validation** screen appears. Proceed to run the Lustre Validation tests.

If any errors occur during this procedure, see *Troubleshoot the Lustre Network Screen*.

11.3 Troubleshoot the Lustre Network Screen

This section contains the causes and possible resolutions for errors that may occur while configuring the Lustre network interfaces for Sonexion 3000 and 3000t.

Carefully inspect any errors generated by CSM. Frequently the errors reveal the exact problem through backend script output, middleware issues, etc.

Some of the procedures require root system access. Contact Cray Support to acquire root access, if needed.

"Expected and actual interface addresses differ, possibly some addresses are already in use" error.

This error is returned after entering network settings and selecting **Next**.

1. The most likely cause is an error in the data entered on this screen. Either a typo was entered or some addresses in the ranges being assigned are already in use.
2. Return to *Configure the Lustre Network Interfaces* on page 88 and re-enter the data, making certain it is correct.
3. If re-entering the data does not work, either change the IP address ranges or change the interface addresses for the existing nodes.

Multiple ranges are supported, and it is possible to exclude conflicting addresses from the range.

CSM hangs for longer than 15 minutes while applying network settings

This hang up has many possible causes, because many backend tools and scripts are involved. To investigate:

1. Run the `ps` command to determine the process ID (PID) of the `beSystemNetConfig.sh` script.

   ```
   [MGMT0]$ ps aux | grep beSystem
   ```

   This command returns a list of currently running processes with a name matching the pattern. Find the output for the beSystemNetConfig.sh script and note the PID, which is shown in the second column.

2. Check the status of the `beSystemNetConfig.sh` script. Run the `watch` command using the PID identified in the previous step:

   ```
   [MGMT0]$ watch pstree -la PID
   ```

   Observe if any of the output changes. If the script hangs on the same step for more than 2-3 minutes, several options are possible:

   - If the script hangs on OST mount, check the network by pinging all IP addresses assigned to the cluster from all hosts (e.g., make certain that the OSS node can be pinged from the MGS IP and vice-versa).
   - If it is a network issue, make certain all cabling and switches are installed correctly.

     If the network looks fine, go to the node hosting the OST and run:

     ```
     [nxxx]$ dmesg | tail -50
     ```

     This lists any kernel errors. If there are no errors related to the OST mount, go to the next step.

   - Examine the `/proc/mdstat` file:

     ```
     [nxxx]$ cat /proc/mdstat
     ```

     Verify that the corresponding MD device is present. If the device is not present, check `dmesg` for `mdraid` assembling problems. Most likely, one of the drives needed for this has failed and must be replaced.

     Some of these procedures require root system access. Contact Cray Support to acquire root access, if needed.

     1. Show the location of the problem array assembly:

        ```
        [nxxx]$ sudo mdraid-activate -d -m
        ```

or

```
[nxxx]$ sudo bash -x mdraid-activate -d -m
```

(The second command provides more information.)

2. Check to see if any hard drives have failed:

```
[nxxx]$ sudo dm_report
```

Review the output for the failed drive (see slot 16 in the sample output below):

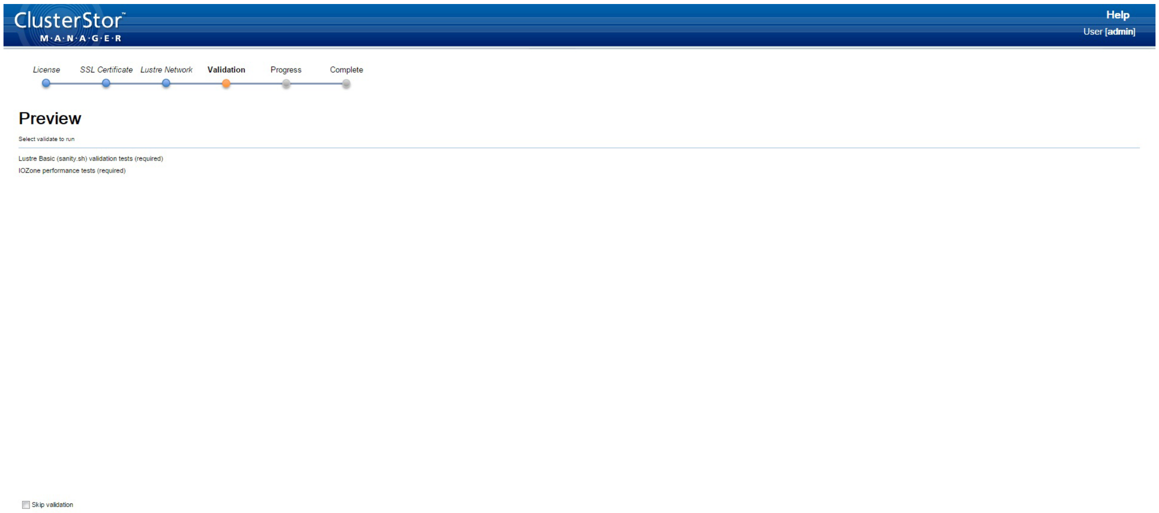
```
[admin@n005 ~]$ sudo dm_report
Diskmonitor Inventory Report: Version: 1.0.x.1.5-16.x.2412 Host: fgbe05
Time: Fri Mar 14 14:10:38 2014

encl: 0, wwn: 50050cc10c4002c8, dev: /dev/sg90, slots: 84, vendor: XYRATEX , product_id: ,
dev1: /dev/sg1 slot: 0,
wwn: 5000cca01b4dff08, cap: 2000398934016, dev: sdbd, parts: 0, status: Foreign Arrays, dev1:
sdej, t10: 11100111120
slot: 1, wwn: 5000cca01b4dbc24, cap: 2000398934016, dev: sdbb, parts: 0, status: Foreign
Arrays,  dev1: sdeh, t10: 11100111120
slot: 2, wwn: 5000cca01c4bfbe8, cap: 2000398934016, dev: sdei, parts: 0, status: Foreign Arrays,
dev1: sdbc, t10: 11100111120
slot: 3, wwn: 5000cca01b38b8a4, cap: 2000398934016, dev: sdek, parts: 0, status: Foreign Arrays,
dev1: sdbe, t10: 11100111120
slot: 4, wwn: 5000cca01c4bd708, cap: 2000398934016, dev: sdef, parts: 0, status: Foreign Arrays,
dev1: sdaz, t10: 11100111120
slot: 5, wwn: 5000cca01c4c5314, cap: 2000398934016, dev: sdeg, parts: 0, status: Foreign Arrays,
dev1: sdba, t10: 11100111120
slot: 6, wwn: 5000cca01c4ade98, cap: 2000398934016, dev: sddx, parts: 0, status: Foreign Arrays,
dev1: sdar, t10  11100111120
slot: 7, wwn: 5000cca01c4af0d0, cap: 2000398934016, dev: sdax, parts: 0, status: Foreign Arrays,
dev1: sded, t10: 11100111120
slot: 8, wwn: 5000cca01c4c32e8, cap: 2000398934016, dev: sdee, parts: 0, status: Foreign Arrays,
dev1: sday, t10  1110011112
slot: 9, wwn: 5000cca01b4ac9a8, cap: 2000398934016, dev: sddy, parts: 0, status: Foreign Arrays,
dev1: sdas, t10 11100111120
slot: 10, wwn: 5000cca01c4aef90, cap: 2000398934016, dev: sdeb, parts: 0, status: Foreign
Arrays, dev1: sdav, t10: 11100111120
slot: 11, wwn: 5000cca01c4c96cc, cap: 2000398934016, dev: sdec, parts: 0, status: Foreign
Arrays, dev1: sdaw, t10: 11100111120
slot: 12, wwn: 5000cca01b4abb54, cap: 2000398934016, dev sddz, parts: 0, status: Foreign Arrays,
dev1: sdat, t10 11100111120
slot: 13, wwn: 5000cca01b4abbc8, cap: 2000398934016, dev: sdea, parts: 0, status: Foreign
Arrays, dev1: sdau, t10: 11100111120
slot: 14, wwn: 5000cca01c4c0284, cap: 2000398934016, dev: sdfl, parts: 0, status: Foreign
Arrays, dev1: sdcf, t10: 11100111120
slot: 15, wwn: 5000cca01c4b9984, cap: 2000398934016, dev: sdfj, parts: 0, status: Foreign
Arrays, dev1: sdcd, t10: 11100111120
slot: 16, wwn: 5000cca01c4c34f4, cap: 2000398934016 dev: sdfk, parts: 0, status: Failed, dev1:
sdce, t10: 11100111120
slot: 17, wwn: 5000cca01b4ce748, cap: 2000398934016, dev: sdfm, parts: 0, status: Foreign
Arrays, dev1: sdcg, t10: 11100111120
slot: 18, wwn: 5000cca01c4c3574, cap: 2000398934016, dev: sdfh, parts: 0, status: Foreign
Arrays, dev1: sdcb, t10: 11100111120
T10_key_begin:
GRD_CHK(1), APP_CHK(1), REF_CHK(1), ATO(1), RWWP(1), SPT(1), P_TYPE(1), PROT_EN(1), DPICZ(1),
FMT(1), READ_CHK(1)
T10_key_end
End_of_report
```

11.4 Run Lustre Validation Tests

About this task

This procedure prepares to run both Lustre basic tests and IOZone performance tests. When the tests complete, the results display on the progress screen and are available for download.

Figure 26. Lustre Validation Tests Screen



Preview

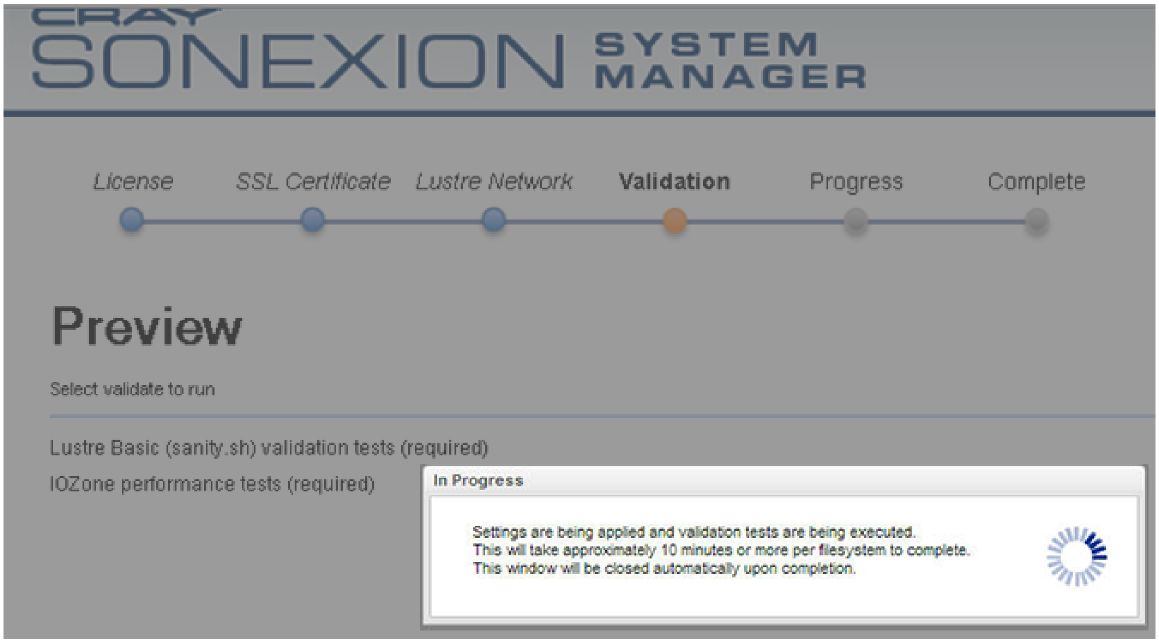
Select validate to run

Lustre Basic (sanity.sh) validation tests (required)

IOZone performance tests (required)

□ Skip validation

Figure 27. Lustre Validation Tests Screen



Preview

Select validate to run

Lustre Basic (sanity.sh) validation tests (required)

IOZone performance tests (required)

In Progress

Settings are being applied and validation tests are being executed.
This will take approximately 10 minutes or more per filesystem to complete.
This window will be closed automatically upon completion.

Procedure

If Lustre was Previously Mounted

1. Unmount the file system before running CSM:

   ```
   [MGMT0]$ cscli unmount -f file_system_name
   ```

   If a `node not unmounted` error occurs, check to see what is mounted:

   ```
   [MGMT0]$ pdsh -a mount | grep -c lustre
   ```

2. SSH into the MGS node (`n002`).

3. Get the resource name for md65:

   ```
   [admin@n002]$ sudo crm_mon -1r
   ```

4. Stop the resource:

   ```
   [admin@n002]$ sudo stop_xyraid nodename_m65_group
   ```

5. Confirm the results:

   > **IMPORTANT:** Do not run the command as `root` user. The command will not reset the password, but asks for a new one. Save the log messages; they will be deleted with each iteration.

   ```
   [admin@n002]$ sudo crm_mon -1r
   ```

   If an error occurs, log into the primary MGMT node and undo the wizard changes:

   ```
   [MGMT0]$ cscli cluster_mode --mode pre-shipment
   ```

6. Click **Next** to launch the tests, and proceed to *Run Lustre Validation Tests -- Lustre Not Mounted* on page 95.

11.5 Run Lustre Validation Tests -- Lustre Not Mounted

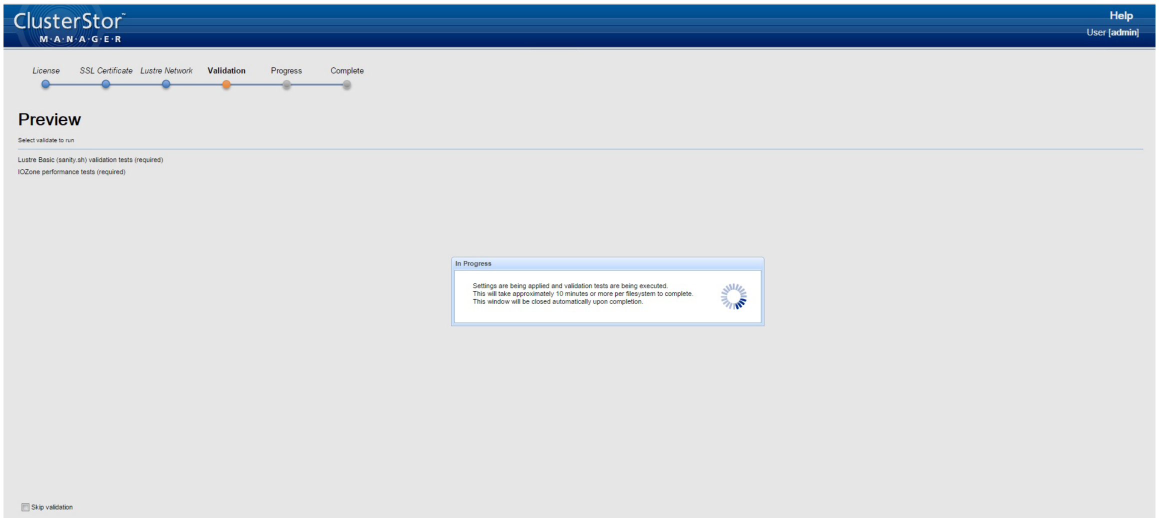
Prerequisites

Lustre was either never mounted on the system or it has been unmounted.

About this task

This procedure runs both Lustre basic tests and IOZone performance tests on a Sonexion 3000 and 3000t system. When the tests complete, the results display on the progress screen and are available for download.

Figure 28. Lustre Validation Tests Progress Screen



Preview

Select validate to run

Lustre Basic (sanity.sh) validation tests (required)

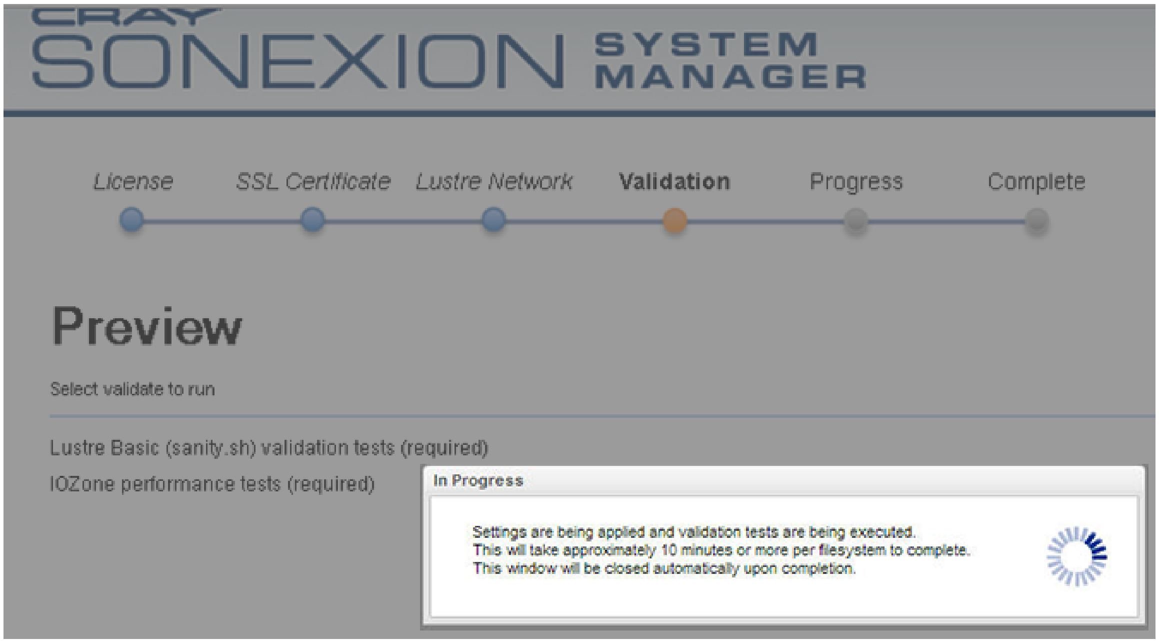
IOZone performance tests (required)

In Progress

Settings are being applied and validation tests are being executed.
This will take approximately 10 minutes or more per filesystem to complete.
This window will be closed automatically upon completion.

☐ Skip validation

Figure 29. Lustre Validation Tests Progress Screen



Preview

Select validate to run

Lustre Basic (sanity.sh) validation tests (required)

IOZone performance tests (required)

In Progress

Settings are being applied and validation tests are being executed.
This will take approximately 10 minutes or more per filesystem to complete.
This window will be closed automatically upon completion.

Procedure

1. Click **Next** at the bottom of the screen to launch the tests.

   The **Validation Test In Progress** screen opens while the validation tests are being performed.

2. Wait approximately 15 minutes per SSU or Server Model 1 cluster for the validation tests to complete.

   The more SSUs (or SSU+ESUs) and Server Model 1 in the system, the more time the tests take. When the tests are complete, the "Progress" screen appears showing the configuration and test procedures to be complete.

11.6 Monitor Progress Screen

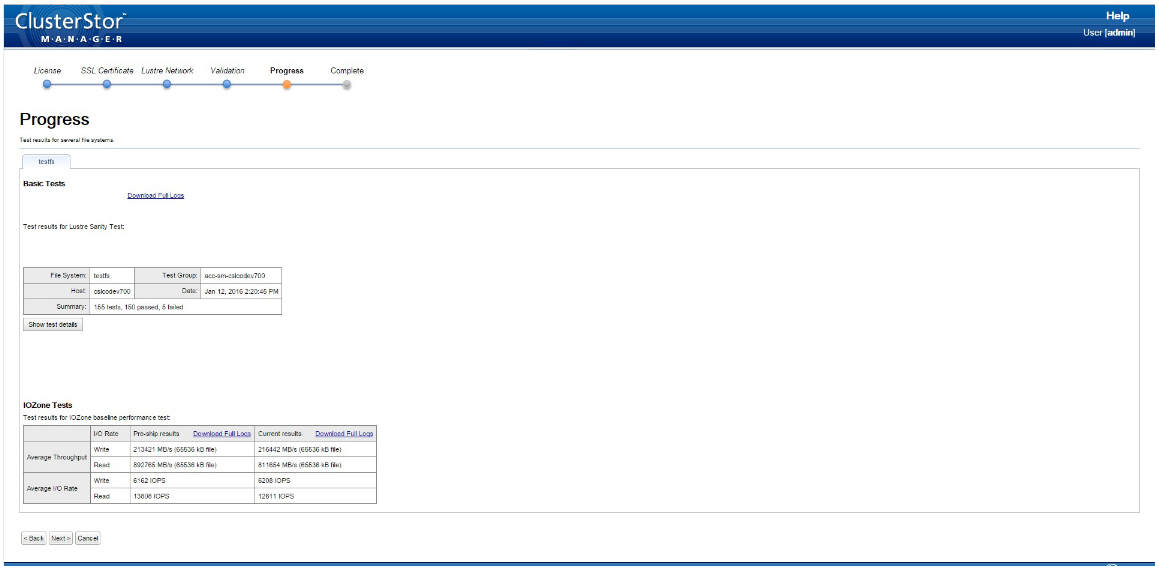
Prerequisites

Lustre validation tests are complete and the results are displayed on the **Progress** screen.

About this task

Once the Lustre validation tests have completed, the results are displayed automatically on the **Progress** screen. Note that if the tests do not run, no I/O rate information will appear in the IOZone Tests section of the **Progress** screen. In this case, see *Post-Install Checkout* on page 102 for advanced procedures.

Figure 30. Progress Screen



Progress

Test results for several file systems.

Basic Tests

Download Full Logs

Test results for Lustre Sanity Test:

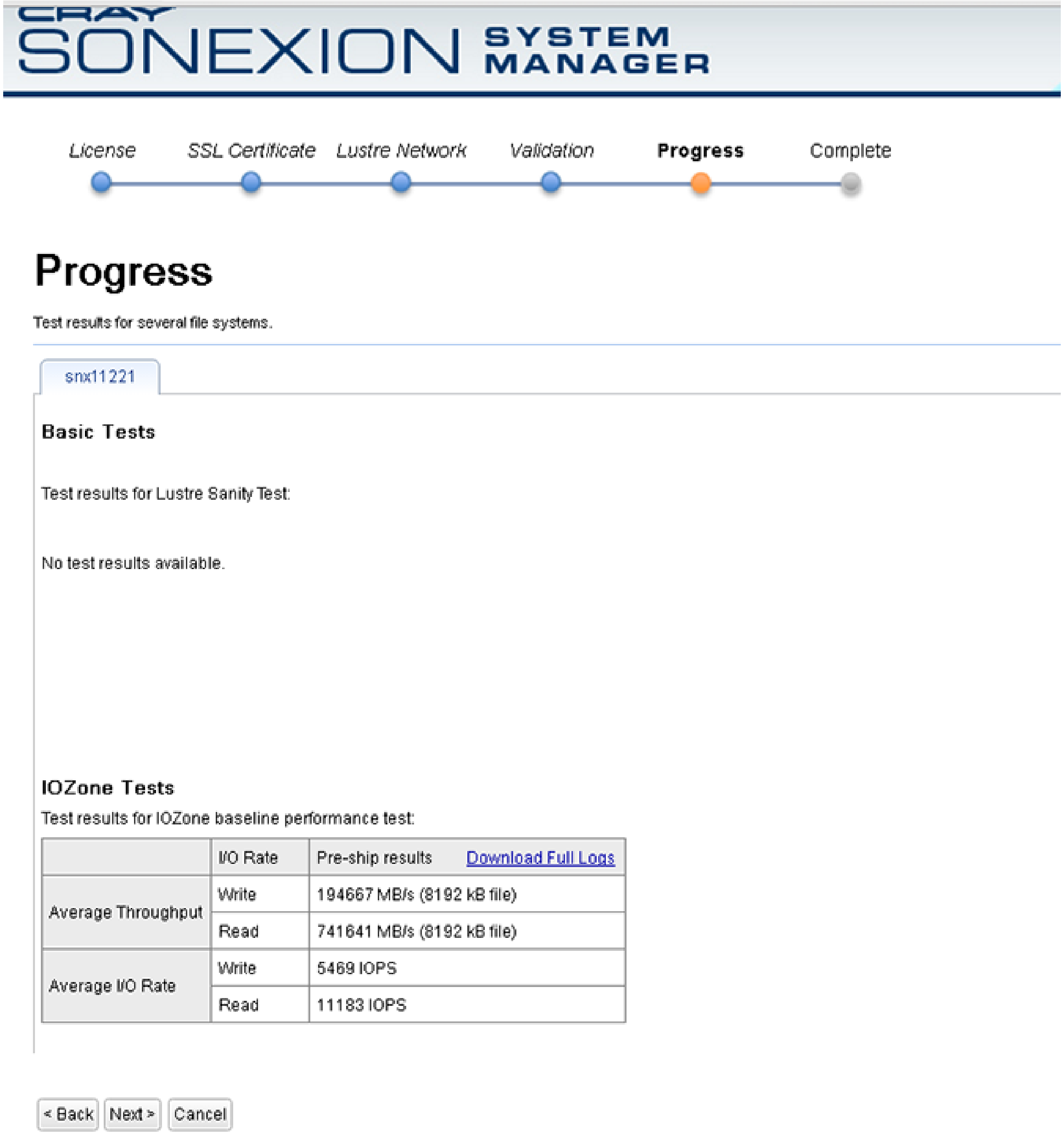
File System:	testfs	Test Group:	acc-sm-cslcodev700
Host:	cslcodev700	Date:	Jan 12, 2016 2:20:45 PM
Summary:	155 tests, 150 passed, 5 failed		

IOZone Tests

Test results for IOZone baseline performance test:

	I/O Rate	Pre-ship results Download Full Logs	Current results Download Full Logs
Average Throughput	Write	213421 MB/s (65536 kB file)	216442 MB/s (65536 kB file)
	Read	892765 MB/s (65536 kB file)	811654 MB/s (65536 kB file)
Average I/O Rate	Write	6162 IOPS	6208 IOPS
	Read	13808 IOPS	12611 IOPS

Figure 31. Progress Screen



Progress

Test results for several file systems.

Basic Tests

Test results for Lustre Sanity Test:

No test results available.

IOZone Tests

Test results for IOZone baseline performance test:

	I/O Rate	Pre-ship results Download Full Logs
Average Throughput	Write	194667 MB/s (8192 kB file)
	Read	741641 MB/s (8192 kB file)
Average I/O Rate	Write	5469 IOPS
	Read	11183 IOPS

Procedure

1. Review the contents of the displayed screen.
2. Click **Next** after reviewing the test results.

   The **Complete** screen displays.

11.7 Review the Complete Screen

Prerequisites

Sonexion configuration has been completed.

About this task

The **Complete** screen confirms the configuration process is complete and the storage cluster is operational.

Procedure

1. To finish the process, click **Continue** to exit the first-run configuration and proceed to CSM.

2. Start the Lustre file system.

   This step can be performed either on the primary MGMT node (via CSCLI) or the CSM GUI.

   a. From the primary MGMT node:

   ```
   [MGMT0]$ cscli mount -f file_system_name
   ```

   Sample output:

   ```
   [admin@dvtrack00 ~]$ cscli mount -f cls12345
   mount: MGS is starting...
   mount: MGS is started!
   mount: cls12345 is started on cls12345n[02-03]!
   mount: cls12345 is started on cls12345n[04-05]!
   mount: cls12345 is started on cls12345n[06-07]!
   mount: File system cls12345 is mounted.
   [admin@cls12345n00 ~]$
   ```

   -OR-

   b. From the CSM GUI, perform the following:

   1. Click the **Node Control** tab.
   2. Select **Nodes in FS** ***fsname*** (which is usually the default).
   3. Click **All Nodes in Filter** menu tab.
   4. Select **Start Lustre** from the drop-down menu.

3. CSM starts Lustre and the Sonexion Cray Sonexion System Manager system is ready for use.

11.8 Specify Lustre User Authentication

Prerequisites

IMPORTANT: Stop Lustre on Sonexion Cray Sonexion System Manager systems before configuring Lustre user authentication.

About this task

This optional procedure may be performed during the first-run configuration or at a later time, as needed. In addition, Lustre user authentication may be performed using the CSM GUI or by using CSCLI commands.

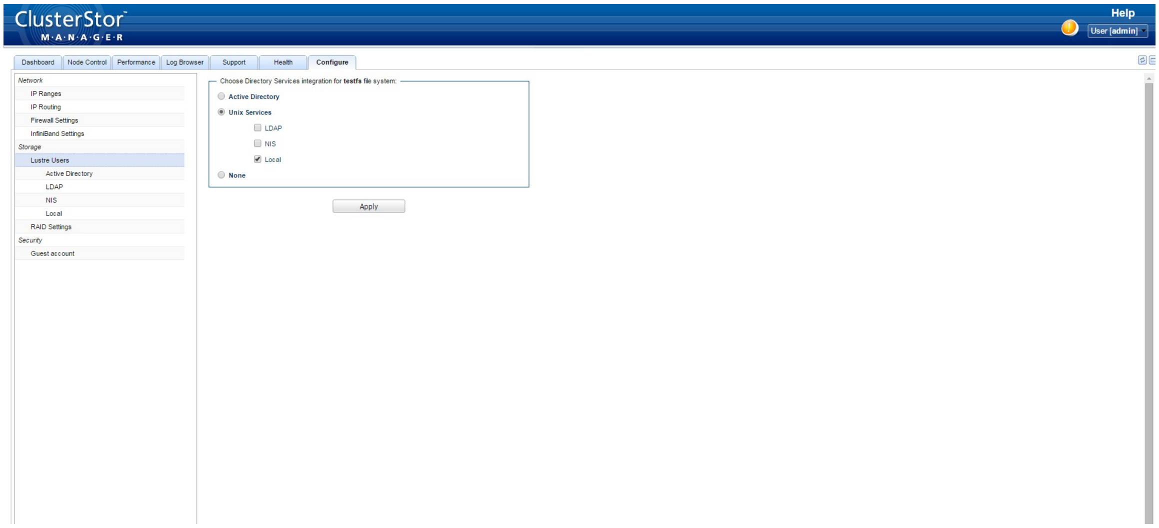
The procedure is used to select and configure commonly used Windows and Unix directory services to authenticate users who need access to the Lustre file system from Lustre clients. Sonexion 3.0.0 systems support the use of Active Directory, LDAP, NIS, and local authentication, and also support:

- Multiple LDAP servers
- Multiple authentication methods simultaneously – NIS, LDAP, etc., (NSS or AD upcall)

The remainder of this topic describes configuring Lustre user authentication using the CSM GUI. Configuration is performed from the CSM **Configure** tab (see figure below). General steps involve:

1. Select the directory services integration to be used.
2. Configure each authentication method.

Figure 32. Lustre Users Screen



Choose Directory Services integration for **testfs** file system:

○ **Active Directory**

● **Unix Services**

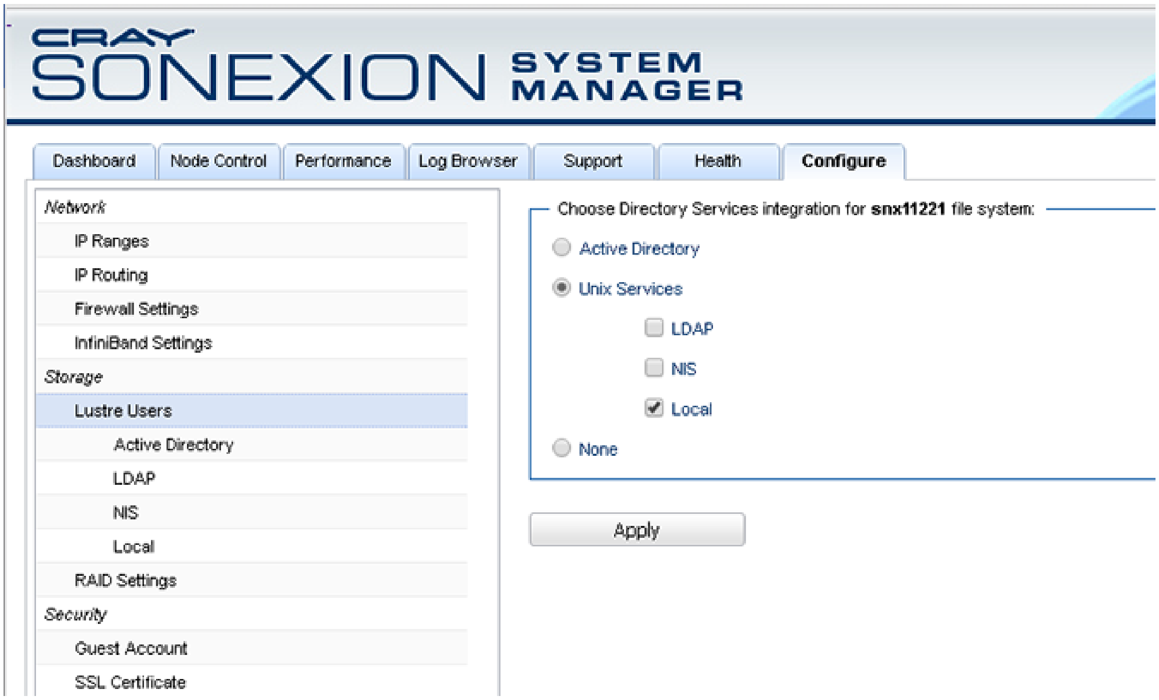
□ LDAP

□ NIS

☑ Local

○ **None**

Figure 33. CSM Lustre Users Screen



Choose Directory Services integration for **snx11221** file system:

○ Active Directory

● Unix Services

□ LDAP

□ NIS

☑ Local

○ None

Procedure

1. Prior to configuring authentication, stop Lustre:

```
[MGMT0]$ cscli unmount -f file_system_name
```

2. On the CSM **Configure** tab select **Lustre Users** from the left side menu.

3. Set Windows and Unix directory services integration for the Lustre file system.

a. Select **Active Directory** for Windows directory services integration.

~OR~

b. Select **Unix Services**, and then select which Unix directory services to use. More than one may be selected. Available options are LDAP, NIS, or Local.

4. Click **Apply**.

5. At this point, configure the individual directory services that were selected above.

For more information, see:

- *Specify Lustre User Authentication via the Active Directory Screen*
- *Specify Lustre User Authentication via the LDAP Screen*
- *Specify Lustre User Authentication via the NIS Screen*
- *Specify Lustre User Authentication via the Local Screen*

11.9 Post-Install Checkout

Prerequisites

Installation of the Sonexion Cray Sonexion System Manager system is complete and the Lustre file system is running.

Procedure

1. If Lustre is not running, start the Lustre file system:

   ```
   [MGMT0]$ cscli mount-f file_system_name
   ```

2. Run `obdfilter-survey` on the primary OSS nodes:

   ```
   [MGMT0]$ sudo pdsh -g oss=primary 'nobjlo=8 nobjhi=8 thrlo=1408 thrhi=1408 \
   size=$((256*1024)) rszmax=$((128*1024)) rszlo=4096 rszhi=4096 \
   obdfilter-survey'
   ```

3. Run the `obdfilter-survey` on the secondary OSS nodes:

   ```
   [MGMT0]$ sudo pdsh -g oss=secondary 'nobjlo=8 nobjhi=8 thrlo=1408 \
   thrhi=1408 size=$((256*1024)) rszmax=$((128*1024)) rszlo=4096 \
   rszhi=4096 obdfilter-survey'
   ```

4. Set the time zone.

   a. Check if the current time zone requires setting:

   ```
   [MGMT0]$ date
   ```

   b. List the available time zones:

   ```
   [MGMT0]$ cscli set_timezone --list
   ```

   c. Set the time zone:

   ```
   [MGMT0]$ cscli set_timezone --set COUNTRY/CITY
   ```

   d. Use the `date` command again to verify the time zone is correct.

11.10 Test Client Mount of Lustre File System

About this task

This procedure describes how to test-mount a client on the Lustre file system for the secondary MGMT node (MGMT1) to verify that the system functions correctly. The steps below describe how to use the CSCLI to mount the Lustre file system on the secondary MGMT node.

Procedure

1. Determine the file system name. On the primary MGMT node:

```
[admin@cls12345n000 ~]$ cscli fs_info
```

Examples show a file system named `cls12345`:

```
[admin@cls12345n000 ~]$ cscli fs_info
----------------------------------------------------------------------------------------------------
OST Redundancy style: Declustered Parity (GridRAID)
Disk I/O Integrity guard (ANSI T10-PI) is not supported by hardware
----------------------------------------------------------------------------------------------------
Information about "cls12345" file system:
----------------------------------------------------------------------------------------------------
Node            Role            Targets         Failover partner        Devices
----------------------------------------------------------------------------------------------------
cls12345n000    mgmt            1 / 0           cls12345n001
cls12345n001    mgmt            1 / 0           cls12345n000
cls12345n002    mgs             0 / 1           cls12345n003             /dev/md65
cls12345n003    mds             0 / 1           cls12345n002             /dev/md66
cls12345n004    oss             0 / 1           cls12345n005             /dev/md0
cls12345n005    oss             0 / 1           cls12345n004             /dev/md1
```

2. If the output in the previous command indicates that the Lustre file system is not started (e.g., the `Targets` column shows `0 / 1` instead of `1 / 1` for OSS nodes), start the Lustre file system:

```
[admin@cls12345n000 ~]$ cscli mount -f cls12345
mount: MGS is starting...
mount: MGS is started!
mount: cls12345 is started on cls12345n002,cls12345n003!
mount: cls12345 is started on cls12345n004,cls12345n005!
mount: All start commands for filesystem cls12345 were sent.
mount: Use "cscli show_nodes" to see mount status.
```

```
[admin@cls12345n000 ~]$ cscli show_nodes
--------------------------------------------------------------------------------
---------
 Hostname      Role     Power State  Service State  Targets  HA Partner    HA
Resources
--------------------------------------------------------------------------------
---------
 cls12345n000  MGMT     On           Started        1 / 0    cls12345n001  Local
 cls12345n001  MGMT     On           Started        1 / 0    cls12345n000  Local
 cls12345n002  MDS,MGS  On           Started        1 / 1    cls12345n003  Local
 cls12345n003  MDS,MGS  On           Started        1 / 1    cls12345n002  Local
 cls12345n004  OSS      On           Started        1 / 1    cls12345n005  Local
 cls12345n005  OSS      On           Started        1 / 1    cls12345n004  Local
--------------------------------------------------------------------------------
---------
```

3. Log in to the secondary MGMT node:

```
[MGMT0]$ ssh -l admin secondary_MGMT_node
```

4. Change to root user on the secondary MGMT node:

```
[MGMT1]$ sudo su -
```

5. Obtain the Lustre network identifier (NIDs) of each MGS node (typically n002 and n003:

```
[MGMT1]# ssh n002 lctl list_nids
```

```
[MGMT1]# ssh n003 lctl list_nids
```

The NIDs will resemble:

```
172.18.1.3@o2ib
```

```
172.18.1.4@o2ib
```

6. Create a directory to use as a temporary mount point for the Lustre client mount:

```
[MGMT1]# mkdir /tmp/testmount
```

7. Mount Lustre to `/tmp/testmount`:

   For example, where the file system is named `cls12345`:

```
[MGMT1]# mount -t lustre 172.18.1.3@o2ib:172.18.1.4@o2ib:/cls12345 /tmp/testmount
```

8. Verify the success of the mount command:

```
[MGMT0]# lfs df -h
```

   The sample command and example output show a successfully mounted Lustre file system.

```
[root@cls12345400 ~]# cscli fs_info
------------------------------------------------------------------------------------
OST Redundancy style: Declustered Parity (GridRAID)
Disk I/O Integrity guard (ANSI T10-PI) is in use ,
------------------------------------------------------------------------------------
Information about "cls12345" file system:
------------------------------------------------------------------------------------
Node Role Targets Failover partner Devices
------------------------------------------------------------------------------------
cls12345402 mgs 0 / 0 cls12345403
cls12345403 mds 1 / 1 cls12345402 /dev/md66
cls12345404 oss 2 / 2 cls12345405 /dev/md0, /dev/md2
cls12345405 oss 2 / 2 cls12345404 /dev/md1, /dev/md3

[root@cls12345400 ~]# ssh cls12345402 lctl list_nids
172.18.1.3@o2ib
[root@cls12345400 ~]#
[root@cls12345400 ~]# mkdir /tmp/testmount
[root@cls12345400 ~]#
[root@cls12345400 ~]# mount -t lustre 172.18.1.3@o2ib:/cls12345 /tmp/testmount
[root@cls12345400 ~]#
[root@cls12345400 ~]# lfs df -h
UUID                  bytes       Used   Available Use% Mounted on
cls12345-MDT0000_UUID     2.8T      4.1G        2.8T   0% /tmp/testmount[MDT:0]
cls12345-OST0000_UUID    84.5T    112.2M       83.7T   0% /tmp/testmount[OST:0]
cls12345-OST0001_UUID    56.5T    108.2M       55.9T   0% /tmp/testmount[OST:1]
cls12345-OST0002_UUID    84.5T    112.2M       83.7T   0% /tmp/testmount[OST:2]
cls12345-OST0003_UUID    56.5T    108.2M       55.9T   0% /tmp/testmount[OST:3]

filesystem summary:   282.0T    440.8M      279.1T   0% /tmp/testmount
```

```
[root@cls12345400 ~]# umount /tmp/testmount
```

9. Remove the temporary mount point:

```
[MGMT1]# rmdir /tmp/testmount
```

   The first-run configuration of the Sonexion system is now complete.

   IMPORTANT: The Lustre file system remains mounted on the cluster. To perform additional updates or changes, unmount the file system first.

10. SSH to MGMT 0 and unmount the file system.

```
[MGMT0]# cscli unmount -f cls12345
```

11.11 Configure SMTP Relay, RAS Email, Remote Support, and Service Console

Prerequisites

Before performing any of the procedures in this section, verify that RAS is installed on the system:

```
[MGMT0]$ cscli service_console show
```

The system responds with the current RAS status, either `Service Console enabled` or `Service Console disabled`.

About this task

This section provides procedures to configure SMTP and email notifications, and to enable/disable the service console. Console-related service configuration changes also can be performed via the CSCLI command-line interface.

In this process, the system administrator configures an external SMTP relay so email notifications pertaining to system events and associated system messages can be issued.

Email notifications are sent to users with email addresses configured by the system administrator. This section explains how to configure a list of users to be sent notifications and how to enable/disable an individual user’s notifications.

> **IMPORTANT:** Email notifications to the specified user list are sent only if the notification functionality is enabled.

The service console login can be enabled (configured) to access guided replacement service procedures, such as replacing a disk. Systems are shipped with service console capabilities enabled, by default.

Procedure

Configure SMTP

1. Show the current SMTP configuration:

```
[MGMT0]$ cscli service_console configure smtp show
SMTP relay: mailrelayus.us.company.com:25
```

The example above shows a hypothetical configuration.

2. Configure the SMTP relay:

```
[MGMT0]$ cscli service_console configure smtp relay \
--host=smtp_host --port=smtp_port
```

where: *smtp_host* (required field) is the SMTP relay host for email notifications, and *smtp_port* is the SMTP relay port (default: 25).

Use the `--help` flag to obtain additional information for configuring the SMTP relay. For example:

```
[MGMT0]$ cscli service_console configure smtp relay --help
```

Configure the Email Notification List

This section explains how to configure a list of users to be sent notifications and how to enable/disable individual user notifications.

> **IMPORTANT:** Email notifications to the specified user list are sent only if the notification functionality is enabled.

3. Display a list of users that are already in the email notification list:

```
[MGMT0]$ cscli service_console configure email show
```

4. Add an email address:

```
[MGMT0]$ cscli service_console configure email add –A user@domain.com
```

Delete an existing email address:

```
[MGMT0]$ cscli service_console configure email delete –A user@domain.com
```

> **IMPORTANT:** Email notifications to the specified user list are sent only if the notification functionality is enabled.

5. Verify that the email address has been correctly established:

```
[MGMT0]$ cscli service_console configure email show
```

6. Enable user email support:

```
[MGMT0]$ cscli service_console configure email enable
```

(To disable user email support):

```
[MGMT0]$ cscli service_console configure email disable
```

7. Send a test email to verify that the email address has been properly added:

```
[MGMT0]$ cscli service_console configure email send_test_email
```

If the SMTP relay is configured properly, the configuration emails are enabled, and the specified email addresses have been set up to receive notifications, the following test email message is sent to the user list:

```
Subject: Remote Service TEST: cls12345
Body:
System Identification:
       System Serial Number: cls12345
       System Identifier:
Current System Time: Tue, 21 Oct 2017 18:05:54
System Timezone: PDT
TEST: This email is to test system SMTP relay setup only, please
ignore.
```

Example of an email successfully sent to remote support:

```
Successfully sent test email to remote support with subject
'Remote Service TEST: cls12345'
```

Example of an email successfully sent to users whose email addresses are properly configured:

```
Successfully sent test email to configured user(s) with subject
'Remote Service TEST: cls12345'
```

Enable or Disable the Service Console

8. Verify that the service console is enabled:

```
[MGMT0]$ cscli service_console enable
```

IMPORTANT: Sonexion are shipped with service console capabilities enabled, by default.

(To disable the service console:)

```
[MGMT0]$ cscli service_console disable
```

11.12 Re-enable the First-Run CSSM

Prerequisites

Verify that the Sonexion system has been fully started and is running in daily mode.

About this task

This section describes how to return the system to pre-shipment mode from daily mode, so CSM can be run. (After CSM has run and completed, the system goes into daily mode, and must be returned to pre-shipment mode to access the configuration.)

Procedure

1. Stop the Lustre file system (if it is running).

```
[MGMT0]$ cscli unmount -f file_system_name
```

2. Log into the primary MGMT node via SSH:

```
[remote]$ ssh -l admin primary_MGMT_node
```

3. Change to root user:

```
[MGMT0]$ sudo su -
```

4. Change to `pre-shipment` mode; on the primary MGMT node:

```
[MGMT0]# cscli cluster_mode --mode pre-shipment
```

5. Clear the SSL certificate entry in the MySQL table:

```
[MGMT0]# mysql t0db -e "DELETE from ssl_certificate"
```

6. Refresh the browser to reload the GUI and verify that the first-run CSM GUI is enabled.

12 Change Network Settings

12.1 Create Custom LNet Configuration for Sonexion

Prerequisites

For a new system, first complete the setup procedures described in the *Sonexion Field Installation Guide*.

About this task

Follow this procedure to create a custom LNet configuration on the Sonexion system while in the "daily mode".

Procedure

1. Log in to the primary management mode.

2. Change to `root`.

   ```
   $ sudo su -
   ```

3. Stop the Lustre file system.

   ```
   # cscli unmount -f file_system_name
   ```

4. Use the following steps to change the `o2ib` index. First, start the MySQL client and connect to the `t0db` database.

   ```
   # mysql t0db
   ```

5. Display the `mgsNID`, `nidFormat`, and `nidIndex` entries.

   ```
   mysql> select * from property where name in ('nidFormat', 'mgsNID', 'nidIndex');
   +-----+--------------------+-----------+---------------------+-----------+
   | id  | context            | name      | value               | attr_type |
   +-----+--------------------+-----------+---------------------+-----------+
   |  22 | snx11000n:beConfig | nidFormat | l%s@o2ib%d          | str       |
   | 106 | snx11000n:beConfig | nidIndex  | 3                   | int       |
   | 109 | snx11000n:beConfig | mgsNID    | lsnx11000n002@o2ib0 | str       |
   +-----+--------------------+-----------+---------------------+-----------+
          3 rows in set (0.00 sec)
   ```

6. Set the `o2ib` index by modifying the `nidIndex` entry.

   ```
   mysql> update property set value=desired_odib_index where name='nidIndex';
   ```

   For example:

```
mysql> update property set value=2 where name='nidIndex';
Query OK, 1 row affected (0.02 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

7. Set the `mgsNID` entry to match the `o2ib` index.

```
mysql> update property set \
value='original_value@o2ibdesired_o2ib_index' \
where name='mgsNID';
```

For example:

```
mysql> update property set value='lsnx11029n002@o2ib0' where name='mgsNID';
Query OK, 1 row affected (0.04 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

8. Verify the changes.

```
mysql> select * from property where name in ('nidFormat', 'mgsNID', 'nidIndex');
+-----+--------------------+-----------+---------------------+-----------+
| id  | context            | name      | value               | attr_type |
+-----+--------------------+-----------+---------------------+-----------+
|  22 | snx11000n:beConfig | nidFormat | l%s@o2ib%d          | str       |
| 106 | snx11000n:beConfig | nidIndex  | 2                   | int       |
| 109 | snx11000n:beConfig | mgsNID    | lsnx11000n002@o2ib0 | str       |
+-----+--------------------+-----------+---------------------+-----------+
       3 rows in set (0.00 sec)
```

9. Close the MySQL session.

```
mysql> quit
```

10. Run `puppet`.

```
# /opt/xyratex/bin/beUpdatePuppet -sa
```

11. Run the `beSystemNetConfig.sh` script on the primary management node and wait for it to finish.

```
# /opt/xyratex/bin/beSystemNetConfig.sh \
-c file_location/lnet.conf -r file_location/routes.conf \
-i file_location/ip2nets.conf clustername
```

12. Verify that the customized LNet configuration has been applied.

a. List the `nids`.

```
# pdsh -g lustre lctl list_nids | sort
```

b. List the nodes and targets.

```
# cscli fs_info
```

13. Start the Lustre file system and wait for the targets to mount on all system nodes.

```
# cscli mount -f file_system_name
```

14. Modify `modprobe.conf` to support IB Aliases.

The following examples intercept the LNet module and setup IB aliases before the module is loaded. **Each example below is a single line** in `lnet.conf`.

a. Add `ib0:1` to `10.10.101` subnet.

```
## Create aliases.
## Intercepts the Lnet module and sets up the aliases before the module is loaded install lnet
```

```
/sbin/ip -o -4 a show ib0 | /usr/bin/awk '/inet/{s=$4;sub("10\\.10\\.100\
\.","10.10.101.",s);print "/sbin/ip address add dev ib0 label ib0:1",s}' | /bin/sh ;/sbin/
modprobe --ignore-install lnet $CMDLINE_OPTS
```

b. Remove all `ib0:1` aliases when LNet module is unloaded.

```
## Remove all ib0:1 aliases when lnet module is unloaded

remove lnet /sbin/modprobe -r --ignore-remove lnet && /sbin/ip -o -4 a show label ib0:1 | awk
'{print "/sbin/ip address del dev ib0 label ib0:1",$4}' | /bin/sh
```

c. Configure a metadata server on a 3rd IPv4 subnet.

```
install lnet if nodeattr mds; then /sbin/ip -o -4 a show ib0 | /usr/bin/awk '/inet/{s=$4;sub("10\
\.10\\.100\\.","10.10.102.",s);print "/sbin/ip address add dev ib0 label ib0:1",s}' | /bin/sh ;
else /sbin/ip -o -4 a show ib0 | /usr/bin/awk '/inet/{s=$4;sub("10\\.10\\.100\
\.","10.10.101.",s);print "/sbin/ip address add dev ib0 label ib0:1",s}' | /bin/sh ; fi ; /sbin/
modprobe --ignore-install lnet $CMDLINE_OPTS
```

12.2 Change DNS Resolver Configuration

About this task

Follow this procedure to change the DNS resolver, the system service that translates URLs into IP addresses.

Procedure

1. `SSH` into the primary MGMT node:

```
$ ssh -l admin primary_MGMT_node
```

2. Sudo to `root`:

```
[MGMT0]$ sudo su -
```

3. Update the DNS settings in the `t0db` database:

```
[MGMT0]$ mysql t0db -e "replace into property(context,name,value,attr_type)
values (\"$(nodeattr -VU cluster):beSystemNetConfig\",\"nameServers\",\
"xx.xx.xx.xx yy.yy.yy.yy\",\"str\")"
```

Where *`xx.xx.xx.xx`* and *`yy.yy.yy.yy`* are the IP addresses of the primary and secondary DNS servers, respectively.

4. Propagate the settings:

```
[MGMT0]$ /opt/xyratex/bin/beUpdatePuppet -s -g mgmt
```

12.3 Change Externally Facing IP Addresses

About this task

Use this procedure to change externally facing IP addresses, for customers to change one or both of the IP addresses of a Sonexion system after it has been installed. Each MGMT node binds an Ethernet interface to one of these externally facing IP addresses. On release 1.5.0 and later, that interface is `eth1`, which is used in the following examples.

Procedure

1. Log in to the secondary MGMT node.

2. SSH to the primary MGMT node:

   ```
   $ ssh -l admin primary_MGMT_node
   ```

3. Sudo to `root`:

   ```
   [admin@n000]$ sudo su -
   ```

4. Edit the Ethernet configuration file and change as desired:

   ```
   [root@n000]# vi /etc/sysconfig/network-scripts/ifcfg-eth1
   ```

   If the system was initially configured to use dynamic (DHCP) IP addresses, the file will look like this:

   ```
   DEVICE=eth1
   BOOTPROTO=dhcp
   ONBOOT=yes
   ```

   If the system was configured to use static IP addresses, it will look like this:

   ```
   DEVICE=eth1
   BOOTPROTO=static
   IPADDR=xx.xx.xx.xx
   NETMASK=255.255.x.x
   GATEWAY=xx.xx.xx.xx
   ONBOOT=yes
   ```

   Where *`xx.xx.xx.xx`* is the valid IP address.

5. Toggle the Ethernet interface:

   ```
   [root@n000]# ifdown eth1; ifup eth1
   ```

6. Exit out of the SSH session for the primary MGMT node.

7. Log in to the primary MGMT node and SSH into the secondary MGMT node.

8. Repeat steps *3* on page 112 through *5* on page 112 to configure the externally facing IP address on the secondary MGMT node.

12.4 Change LDAP Settings in Daily Mode

About this task

Use this procedure to change the LDAP settings, which are stored in the t0db database table ldap_setup. The following columns are present in this table and are used to configure LDAP on Sonexion nodes:

Table 5. LDAP Settings

Setting	Use
server_name	The LDAP server
port	The port that the LDAP server listens on (typically 389)
base_dn	The base DN to search
user_dns	Search patch for user information
group_dns	Search path for group information
bind_dns	DN to bind to the LDAP directory
password	Password to use with bind_dn

To change the LDAP settings of a running cluster, it is necessary to change the corresponding field with the update MySQL command and then run beUpdatePuppet -sa. The following example shows how to change the base DN to search.

> **IMPORTANT:** As of SU10, the preferred method is to use the cscli lustre_users ldap commands, which are described in the CSCLI reference section or use the CSM GUI, **Configure** tab **LDAP** settings window.

Procedure

1. SSH into the primary MGMT node.

```
$ ssh -l admin primary_MGMT_node
```

2. Sudo to root:

```
[MGMT0]$ sudo su -
```

3. Print the existing configuration:

   a. List the contents of /etc/nslcd.conf.

```
[MGMT0]# cat /etc/nslcd.conf
#
# LDAP Trinity
#
# Auto generated by puppet
# Do not change it manually
#

timelimit 120
bind_timelimit 120
idle_timelimit 3600

# Workaround for names <3 char length. see TRT-1832
validnames /^[a-z0-9._@$][a-z0-9._@$ \~-]*[a-z0-9._@$~-]$/i

#pam_password md5
#bind_policy soft
#ldap_version 3

uri ldap://172.30.12.19:389/
base dc=datacenter,dc=cray,dc=com
base passwd ou=People,dc=datacenter,dc=cray,dc=com
```

b. List the LDAP settings in the `t0db` database table `ldap_setup`.

```
[MGMT0]#  mysql t0db -e "select * from ldap_setup;"
```

```
+---------------+------------------------------+---------+----------+------------+----------+------------+-------------+---------+
| ldap_setup_id | base_dn                      | bind_dn | password | cluster_id | tls_cert | tls_pvtkey | tls_ca_cert | mapping |
+---------------+------------------------------+---------+----------+------------+----------+------------+-------------+---------+
|             1 | dc=datacenter,dc=cray,dc=com | NULL    | NULL     |          1 | NULL     | NULL       | NULL        | default |
+---------------+------------------------------+---------+----------+------------+----------+------------+-------------+---------+
```

4. Change the `base_dn` setting:

```
[MGMT0]# mysql t0db -e "update ldap_setup set
base_dn='dc=new_ldap,dc=example,dc=com'"
```

Where *`new_ldap`* is the new LDAP server.

5. Update `puppet`:

```
[MGMT0]# /opt/xyratex/bin/beUpdatePuppet -sa
```

6. Repeat step 3 and check for the new value(s).

12.5 Configure LDAP over TLS in Daily Mode

About this task

Set up an LDAP/TLS server.

Procedure

1. In the file `/etc/puppet/modules/ldap/templates/ldap.conf.erb`:

a. Add the line:

```
tls_cacert /etc/openldap/cacerts/ca.crt
```

or whatever file has the CA certificate.

b. Change the line:

```
uri ldap://<%= system['serverName'] %>:<%= system['serverPort'] %>/
```

to:

```
uri ldaps://<%= system['serverName'] %>:<%= system['serverPort'] %>/
```

2. In the file `/etc/puppet/modules/ldap/templates/nslcd.conf.erb`:

a. Add the line:

```
tls_cacertfile /etc/openldap/cacerts/ca.crt
```

or whatever file has the CA certificate.

b. Change the line:

```
uri ldap://<%= system['serverName'] %>:<%= system['serverPort'] %>/
```

to:

```
uri ldaps://<%= system['serverName'] %>:<%= system['serverPort'] %>/
```

3. Put the CA certificate file in the appropriate `%=_system['serverName']_%` directory in the image on **n001**:

```
/mnt/nfsdata/images/2.0.0-51/appliance.x86_64/etc/openldap/cacerts/
```

Additional Customer LDAP Parameters

If your customer's LDAP server requires parameters not included in the procedure above, use the following steps to add these changes to the file's `/etc/puppet/modules/ldap/templates/ldap.conf.erb` and `/etc/puppet/modules/ldap/templates/nslcd.conf.erb`.

NOTE: The customer parameter can be added anywhere in the two files except between "`<%`" and "`%>`".

Example: The customer-secure LDAP server requires the following parameter setting:

```
TLS_REQCERT never
```

4. Log in to MGMT node n000 as root .

5. Add the parameter to the following files, noting that it can be added anywhere in the two files except between "`<%`" and "`%>`":

- `/etc/puppet/modules/ldap/templates/ldap.conf.erb`
- `/etc/puppet/modules/ldap/templates/nslcd.conf.erb`

6. Once the parameter is added, puppet configuration needs to be updated.

On MGMT node n000 as root, execute the following commands:

```
pdsh -g mgmt puppetd -tv
pdsh -g mds puppetd -tv
```

7. After puppet configuration has been updated login to the servers n000, n001, n002, n003 and verify that `/etc/ldap.conf and /etc/openldap/ldap.conf` files have been modified as expected.

8. To verify LDAP functionality, execute the following command from each node n000, n001, n002, n003.

```
L_GETIDENTITY_TEST=1 /usr/sbin/l_getidentity_nss snx11168-MDT0000 uid
```

12.6 Configure NIS Support in Daily Mode

Prerequisites

- NIS must not have been previously configured during Site Configuration mode. If it is already configured, see *Change NIS Settings in Daily Mode* on page 117.
- Both the NIS Domain name and the IP address of NIS servers reachable (pingable) from the Management Nodes must be known.

About this task

This procedure applies to releases 1.3.1 and later. Sonexion 1.3.1 supports LDAP and NIS but was intended only to support configuration of NIS during the Customer Wizard phase of the installation.

Procedure

1. Enable support for NIS:

```
[root@n000]# mysql t0db -e 'update filesystem set lustre_upcall="nis"'
```

2. Configure the IP address(es) of NIS servers:

```
[root@n000]# mysql t0db -e 'insert into property (context, name, value,
attr_type) values ("lustre:upcall", "nis_server", "xx.xx.xx.xx","str")'
```

To specify more than one IP address, ensure that they are separated by a single space.

3. Configure the name of the NIS domain:

```
[root@n000]# mysql t0db -e 'insert into property (context, name, value,
attr_type) values ("lustre:upcall", "nis_domain", "xxxxxxxxxx", "str")'
```

Where *`xxxxxxxxxx`* is the value of `nisdomainname` on the relevant server.

4. Update `puppet`:

```
[root@n000]# beUpdatePuppet -sa
```

5. Check that /etc/puppet/data/CSSM/nis.yaml has been updated on the primary and secondary management nodes and contains the following lines (example):

```
[root@n000]# pdsh -a cat /etc/puppet/data/CSSM/nis.yaml 2>/dev/null | dshbak -c
----------------
MGMT[00-01]
----------------
lustre_nis:
nis_domain: xxxxxxxxxx
nis_server: [xx.xx.xx.xx]
```

6. Verify that /etc/yp.conf on the MMU nodes contains the same information (example):

```
[root@n000]# pdsh -a cat /etc/yp.conf 2>/dev/null | dshbak -c
----------------
MGMT[00-03]
----------------
#
#
# CSSM Lustre NIS
#
#
# Auto generated by puppet
# Do not change it manually
#
domain xxxxxxxxxx server xx.xx.xx.xx
```

7. Run these final checks on all MMU nodes:

```
[root@n000]# service ypbind status
[root@n000]# ypwhich
[root@n000]# ypwhich -m
```

12.7 Change NIS Settings in Daily Mode

Prerequisites

NIS has previously been configured, either during customer wizard or following the instructions in *Configure NIS Support in Daily Mode* on page 116.

About this task

Use this procedure, which applies to releases 1.3.1 and later, to change NIS settings using the CSCLI interface in Daily Mode.

Procedure

1. SSH into the primary MGMT node.

```
$ ssh -l admin primary_MGMT_node
```

2. Sudo to root:

```
[admin@n000]$ sudo su -
```

3. Print the existing configuration:

   ```
   [root@n000]# service ypbind status
   [root@n000]# ypwhich
   [root@n000]# ypwhich -m
   [root@n000]# cat /etc/yp.conf
   [root@n000]# mysql t0db -e 'select * from property where name = "nis_domain" or
   name = "nis_server"'
   ```

   Sample output:

   ```
   [root@snx11000n000 ~]# pdsh -g mgmt,mds ypwhich | dshbak -c
   ----------------
   snx11000n[000-003]
   ----------------
   172.30.74.10

   [root@snx11000n000 ~]# pdsh -g mgmt,mds ypwhich -m | dshbak -c
   ----------------
   snx11000n[000-003]
   ----------------
   auto_sw_linux_cf ra.us.cray.com
   auto_sw_linux_sea ra.us.cray.com
   auto_users ra.us.cray.com
   auto_master_linux_mh ra.us.cray.com
   auto_master_linux_cf ra.us.cray.com
   ```

4. Change to IP address of the NIS Server:

   ```
   [root@n000]# mysql t0db -e 'update property set value =
   "xx.xx.xx.xx yy.yy.yy.yy" where name = "nis_server"'
   ```

   One or more IP addresses can be specified for NIS master and its reachable NIS slaves.

5. Change the NIS domain name of the NIS server.

   ```
   [root@n000]# mysql t0db -e 'update property set value = "xxxxxxxxxx" where name
   = "nis_domain"'
   ```

6. Update puppet:

   ```
   [root@n000]# beUpdatePuppet -sa
   ```

   This is required after the previous changes.

7. Repeat step *3* on page 118 to verify the new NIS server settings.

13 Change the NTP Configuration

About this task

Use this procedure to change the Network Time Protocol (NTP) configuration on a Sonexion system without re-running the customer wizard. (Since the wizard makes many changes, it can introduce unnecessary complexity.) The following procedure has these characteristics:

- A change in the **t0db** MySQL database is required.
- As with any direct changes to the database, the database is first backed up.
- Changes are performed in a script session.
- It is best if there are not large differences between the old and new server times.

Procedure

1. Back up the database:

   ```
   # mysqldump t0db > t0db.bak
   ```

2. Determine what is currently in use:

   ```
   # mysql t0db -e "select * from property where name = 'timeServers';"
   ```

3. Update the database:

   ```
   # mysql t0db -e "update property set value = 'list' where name =
   'timeServers';"
   ```

   Where *list* is a space-separated list of timeserver IPs.

4. Propagate changes:

   ```
   # beUpdatePuppet -sa
   ```

14 Check File Systems Using `e2fsck` Command

About this task

Perform the following steps on the Sonexion filesystem. In order to run `e2fsck` on a device or set of devices, the device must not be mounted. In a Lustre environment, even a read-only `e2fsck` runs incorrectly on a mounted and running device.

Procedure

1. Stop Lustre by running the following on the MGMT node:

   ```
   [root@n000]# cscli unmount -f fs_name
   ```

2. SSH to the OSS node containing the devices where the `e2fsck` is to be run, **n004** in this example:

   ```
   [root@snx11000n000]# ssh snx11000n004
   [root@snx11000n004]#
   ```

3. Use the following steps to build the disk arrays to be accessible as mountable devices. The simplest method is to build all the arrays on the system. A Sonexion using MDRAID has four devices; a Sonexion using GridRAID has only one device.

   ```
   [root@snx11000n004]# mdraid-activate -md
   mdadm: /dev/md/snx11000n004:md132 has been started with 2 drives.
   mdadm: /dev/md/snx11000n004:md133 has been started with 2 drives.
   mdadm: /dev/md/snx11000n004:md2 has been started with 10 drives.
   assembled snx11000n004:/dev/md2 in 1 tries
   mdadm: /dev/md/snx11000n004:md140 has been started with 2 drives.
   mdadm: /dev/md/snx11000n004:md141 has been started with 2 drives.
   mdadm: /dev/md/snx11000n004:md6 has been started with 10 drives.
   assembled snx11000n004:/dev/md6 in 1 tries
   mdadm: /dev/md/snx11000n004:md128 has been started with 2 drives.
   mdadm: /dev/md/snx11000n004:md129 has been started with 2 drives.
   mdadm: /dev/md/snx11000n004:md0 has been started with 10 drives.
   assembled snx11000n004:/dev/md0 in 1 tries
   mdadm: /dev/md/snx11000n004:md136 has been started with 2 drives.
   mdadm: /dev/md/snx11000n004:md137 has been started with 2 drives.
   mdadm: /dev/md/snx11000n004:md4 has been started with 10 drives.
   assembled snx11000n004:/dev/md4 in 1 tries
   ```

   In the above output, the OST devices are named `/dev/md#` in the lines beginning `assembled snx11###n###:md# in 1 tries`. From the example above, the device of interest is this:

   ```
   assembled snx11000n004:md2 in 1 tries
   ```

   The other drives are journaling and backup devices.

4. Confirm that OST devices are not mounted, by running `mount` and verifying that they are not present. Only four devices should be mounted for MDRAID systems, or one for GridRAID systems. The following output indicates they are **not** mounted:

```
[root@snx11000n004]# mount
none on /proc type proc (rw)
none on /dev/shm type tmpfs (rw)
none on /tmp type tmpfs (rw)
none on /mnt type tmpfs (rw,size=512k)
none on /WIBS type tmpfs (rw,size=512k)
none on /data type tmpfs (rw,size=512k)
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw)
/dev/sda2 on /mnt/persistent_ha_data type ext4 (rw)
/mnt/persistent_ha_data/dirs/etc/ha.d on /etc/ha.d type none (rw,bind)
/mnt/persistent_ha_data/dirs/var/lib/heartbeat on /var/lib/heartbeat type none (rw,bind)
/mnt/persistent_ha_data/dirs/var/lib/mdraidscripts on /var/lib/mdraidscripts type none (rw,bind)
/mnt/persistent_ha_data/dirs/var/lib/pengine on /var/lib/pengine type none (rw,bind)
/dev/md132 on /WIBS/snx11000n004:md2 type ext2 (rw)
/dev/md128 on /WIBS/snx11000n004:md0 type ext2 (rw)
/dev/md140 on /WIBS/snx11000n004:md6 type ext2 (rw)
/dev/md136 on /WIBS/snx11000n004:md4 type ext2 (rw)
```

For comparison, the following output indicates an **incorrect condition**, in that the devices **md0**, **md2**, **md4**, and **md6** are mounted.

```
[root@snx11000n004]# mount
none on /proc type proc (rw)
none on /dev/shm type tmpfs (rw)
none on /tmp type tmpfs (rw)
none on /mnt type tmpfs (rw,size=512k)
none on /WIBS type tmpfs (rw,size=512k)
none on /data type tmpfs (rw,size=512k)
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw)
/dev/sda2 on /mnt/persistent_ha_data type ext4 (rw)
/mnt/persistent_ha_data/dirs/etc/ha.d on /etc/ha.d type none (rw,bind)
/mnt/persistent_ha_data/dirs/var/lib/heartbeat on /var/lib/heartbeat type none (rw,bind)
/mnt/persistent_ha_data/dirs/var/lib/mdraidscripts on /var/lib/mdraidscripts type none (rw,bind)
/mnt/persistent_ha_data/dirs/var/lib/pengine on /var/lib/pengine type none (rw,bind)
/dev/md132 on /WIBS/snx11000n004:md2 type ext2 (rw)
/dev/md128 on /WIBS/snx11000n004:md0 type ext2 (rw)
/dev/md140 on /WIBS/snx11000n004:md6 type ext2 (rw)
/dev/md136 on /WIBS/snx11000n004:md4 type ext2 (rw)
/dev/md2 on /data/snx11000n004:md2 type lustre (rw,errors=panic,journal_checksum)
/dev/md0 on /data/snx11000n004:md0 type lustre (rw,errors=panic,journal_checksum)
/dev/md6 on /data/snx11000n004:md6 type lustre (rw,errors=panic,journal_checksum)
/dev/md4 on /data/snx11000n004:md4 type lustre (rw,errors=panic,journal_checksum)
```

5. Run the `e2fsck` command as follows. A basic `e2fsck` task is to run a read-only check of the device to diagnose any errors, such as a simple non-invasive action on **md2**.

   Run the command in the form:

```
[root@nxxxx]$ e2fsck -nvf device_to_be_checked
```

   Example:

```
[root@snx11000n004]# e2fsck -nvf /dev/md2
```

   Analysis of `e2fsck` output is described elsewhere. If the output from the above commands indicates the need for a read/write `e2fsck`, use the same command with the appropriate options.

6. Deactivate the RAID to clean up and return the system to a state ready to mount for normal Lustre operation:

```
[root@snx11000n004]# mdraid-deactivate -md
```

15 Disk Watching Daemon

This topic explains the use of the Disk Watching Daemon (DWD) to monitor hard drives in the system.

- **Prerequisites**

  For Sonexion 1.3.1 systems, System Update 019 (SU-019) must be installed for DWD to be available on the system.
- **Service Interruption Level**: Live (can be applied to a live system with no service interruption)
- **System Access Requirements:**

  This procedure is written for use by an admin user and does not require root (super user) access. It is recommended that the technician log in as an administrative user and perform the procedure using CSCLI commands as written, and not log in as root or perform the procedure as a root user.

DWD is a daemon on Sonexion systems that monitors hard drives for symptoms that suggest impending failure. DWD periodically polls all hard drives that are part of a RAID array, and if it finds a symptomatic drive, DWD takes corrective action.

DWD is available on Sonexion systems beginning with software release 1.3.1-019. Note that 1.3.1 systems must have at least System Update 019 (SU-019) installed.

DWD runs on all nodes of a Sonexion system, and monitors every disk drive for three specific failure conditions:

- An excess of timed-out SCSI commands
- Failure of the SMART status check
- Failure during a drive interrogation

DWD Configuration

DWD can be configured to take one of two actions when a drive fails a DWD health check:

- Log the test failure with a suggestion that the drive be removed.
- Log the test failure and then power down the drive, causing the hard drive to be removed from any RAID arrays of which it is a member. After the drive is removed from the RAID array, a replacement hot-spare drive is automatically recruited into the RAID device (for MDRAID configurations only) and a RAID rebuild is triggered. For GridRAID configurations, the drive is removed and a repair operation is triggered to use the built-in hot-spare space of the array.

Even if DWD is configured for this option, the drive is not powered off if the array is already degraded. Instead, DWD reports in syslog:

```
"Array array_name is degraded or without redundancy, no automatic purging".
```

To determine what corrective action DWD will take when a drive fails a health check, run:

```
[admin@n000]$ cscli dwd
```

Sample command and output:

```
[admin@snx11000n000 ~]$ cscli dwd
------------------------------------------------------------
Node Drive power-off on failure Task abortion rate
------------------------------------------------------------
global 0 1
```

In the sample output, the column labeled Drive power-off on failure indicates what action DWD takes when that drive fails one of the health checks.

- A value of 0 (the default) indicates that the drive will not be powered off. The only action that occurs is that a message will be sent to syslog stating that, "You should be Failing out problematic drive."

  For example:

  ```
  Nov 26 18:24:02 cstor01n04 dwd: You should be Failing out problematic drive :
  </sbin/mdadm /dev/md4 --fail /dev/sdo>
  ```

- A value of 1 indicates that DWD will log "Failing out problematic drive" to syslog:

  ```
  Nov 26 18:24:02 cstor01n04 dwd: Failing out problematic drive : </sbin/
  mdadm /dev/md4 --fail /dev/sdo>
  ```

DWD then powers off the drive.

DWD's Drive power-on can be set on failure by using the `cscli dwd -l` command, where `-l` is lethal. For example:

```
[admin@snx11000n000 ~]$ cscli dwd -l yes
dwd: done
[admin@snx11000n000 ~]$ cscli dwd
------------------------------------------------------------
Node Drive power-off on failure Task abortion rate
------------------------------------------------------------
global 1 1
```

At the time of publication, once the "Drive power-off on failure" value is 1, it cannot be set back to 0.

DWD Check 1: Excess Timed-out SCSI Commands

When a disk drive does not respond to an I/O command within 100 seconds, that command is considered failed and the SCSI subsystem attempts to abort it. DWD watches the count of timed-out commands and considers a drive failed after it exceeds a certain number of aborted tasks per (rolling) hour.

This "Task abortion rate" threshold is viewable in CSCLI with the command `cscli dwd`:

Sample command and output:

```
[admin@snx11000n000 ~]$ cscli dwd
------------------------------------------------------------
Node Drive power-off on failure Task abortion rate
------------------------------------------------------------
global 0 1
[admin@snx11000n000 ~]$
```

When the value for the task abortion rate is 1, the actual threshold is automatically set to a value of (queue depth) * 2 + 1, which works out to either 5 or 9, depending on the queue depth setting on the hard drive.

To set the task abortion rate manually, run:

```
[admin@n000]$ cscli dwd -a value
```

Sample command and output:

```
[admin@snx11000n000 ~]$ cscli dwd -a 20
dwd: done
[admin@snx11000n000 ~]$ cscli dwd
------------------------------------------------------------
Node Drive power-off on failure Task abortion rate
------------------------------------------------------------
global 0 20
```

DWD Check 2: Failure of the SMART Status Check

Periodically, DWD runs a SMART inquiry, `smartctl -q -H device`, of all drives that are members of a RAID array assembled on the local host. If the SMART status check returns DISK FAILING, the test fails.

Currently, there are no adjustable parameters for this check. A future release of DWD may allow the user to change the test rate (which is currently hardcoded to occur once a day) or to disable this test.

DWD Check 3: Failure During a Drive Interrogation

If a disk drive accumulates 12 SCSI errors within 1 hour, it is subjected to an interrogation. DWD issues the following commands against the drive:

```
mdadm --examine drive
smartctl -l error drive
```

The test fails if the above commands indicate problems with the drive, either because the command timed out or because it returned errors.

At the time of publication, no drive has failed interrogation. This test is included in DWD for completeness, but success on this test should not be assumed to indicate good disk drive health.

No parameters for this check can be set through the `cscli dwd` command.

16 Support Bundles

Support bundles are collections of event logs from field systems that are collected by Cray personnel and used to debug many Sonexion problems. Sonexion provides a mechanism for collecting support bundles that can be initiated manually or triggered automatically by certain events (e.g., Lustre bugs or failover events). These support bundles should be provided to Cray personnel in the course of requesting technical support.

When a problem is encountered, use the following tabs in the CSSM GUI to assist with resolving the issue:

- The **Health** tab displays details of the host or service alerts and notifications to determine the issues.
- The **Log Browser** tab is used to review the log files for help identifying and diagnosing the issues.
- The **Support** tab shows diagnostic information from the storage cluster, including logs and configuration settings.

Support File Overview

The CSSM **Support** tab provides support functionality for collecting diagnostic information, including logs and configuration settings, on an automatic or manual basis. When a Lustre error occurs, the system automatically collects diagnostic information. Alternatively, Sonexion users can manually collect a diagnostic payload and browse the contents.

The three principal resources for debugging an issue are support bundles, system logs, and GEM logs.

Collection of Sonexion Data in Support Files

When a Lustre error or a system event (such as failover) occurs, Sonexion automatically triggers a process to collect system data and diagnostics, and bundle them in support files. The process waits two minutes before collecting the data to ensure that all consequences of the events and errors are logged. Only one collection process is active at a time.

Multiple errors do not trigger the collection of additional data if the current process is still running or within a two-hour window after the current process was triggered. For example, if a Lustre error occurs at 8:00, triggering data to be collected in support file bundle and the same error occurs one hour later at 9:00, Sonexion does not start a second data collection process related to the later error.

Contents of Support Bundles

Data related to system errors is collected in files, which are packaged together into support bundles. A support bundle is a standard UNIX-compressed file (`tar-gzip`), with files that include:

- System logs for all nodes for the 45-minute period before the error occurred
- List of all cluster nodes and information for each node:
  - Software version
  - Linux kernel and patches

  - Sonexion RPMs
  - OSTs mounted on the node
  - Power states
  - Resource states
  - Relevant processes
  - **Sysrq** data
- Current Apache/WSGI logs from the MGS/MDS
- Application state data (MySQL database dump)
- Diagnostic and performance test logs

Automatic vs. Manual Data Collection

When an error occurs, data collection and the bundling of support files is triggered automatically, and Sonexion users cannot terminate or cancel the operation. Alternately, a user can manually start data collection and create a support bundle. Unlike the automatic process, a manual data collection operation can be canceled.

An operator can also manually start a support bundle collection or import a support bundle, during which time the operator is prompted to select the nodes (defaults to "all") and a window of time (the default is 45 minutes) for logs. After a confirmation dialog appears and is acknowledged, the process begins immediately, there is no 2-minute wait.

16.1 Collect Support Files Manually

About this task

The Sonexion user can manually start system data collection and create support files as described in this procedure. Unlike the automatic process, the manual data collection operation can be canceled.

Procedure

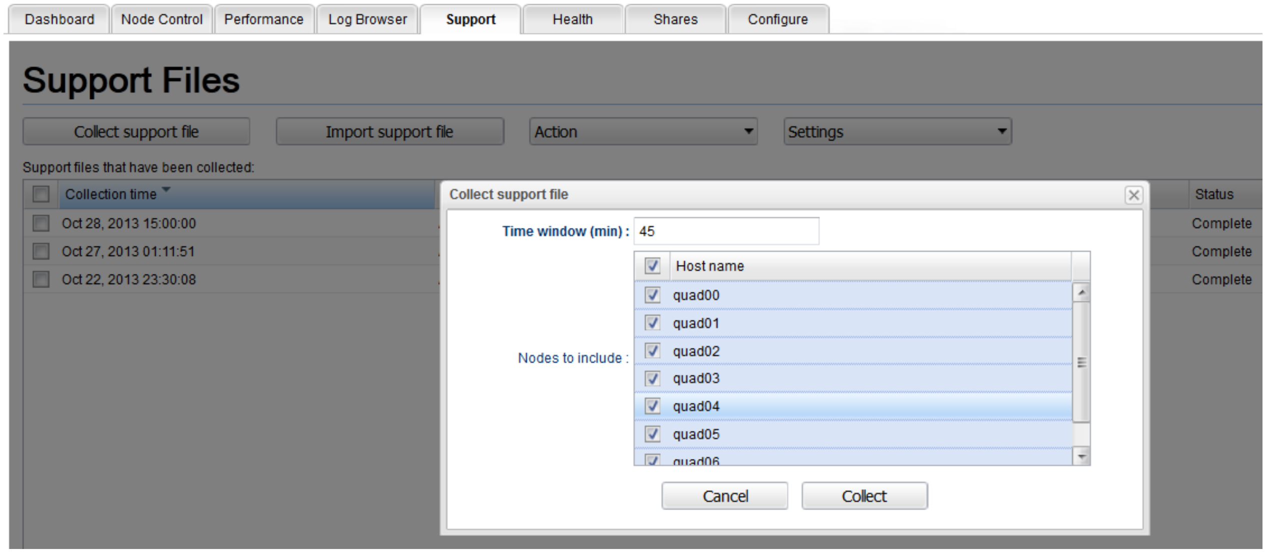
1. Click the CSSM **Support** tab.

   The **Support Files** screen displays.

2. Click the **Collect support file** button.

   The **Collect support file** dialog window opens and lists all nodes in the cluster.

Figure 34. Support Files Screen: Collect File



Support files that have been collected:

□	Collection time	Status
□	Oct 28, 2013 15:00:00	Complete
□	Oct 27, 2013 01:11:51	Complete
□	Oct 22, 2013 23:30:08	Complete

Collect support file

Time window (min) : 45

Nodes to include :

☑	Host name
☑	quad00
☑	quad01
☑	quad02
☑	quad03
☑	quad04
☑	quad05
☑	quad06

3. Specify the data collection parameters for the support file:

   a. Select the **time period** to look back for syslog data to be collected. The default is 45 minutes.

   b. Select the **nodes** for which data will be collected (the check box next to **Hostname** selects all nodes).

   c. Click the **Collect** button.

      The data collection process starts using the specified parameters. While collecting data, it displays in the status field “Still collecting, *xx*% complete”. When it is complete, it states “Done.” To terminate the operation at any point, click **Cancel**.

      When the operation is complete, the support file, which is a Zip file containing hundreds of different log files, is created.

16.2 Create Support Bundles Using CLI

About this task

Support bundles can be created manually CLI commands. For further information, see *support_bundle Command* on page 227.

Procedure

1. Log into the primary MGMT node via `SSH`.

```
$ ssh -l admin primary_MGMT_node
```

2. Change to `root` user.

```
$ sudo su -
```

3. Collect the support bundle.

   - To collect the bundle using the default 45 minute time period:

     ```
     [root@n000] cscli support_bundle -c
     ```

   - To collect the bundle using a different time period, *minutes*:

     ```
     [root@n000] cscli support_bundle -c -t minutes
     ```

   For example:

   ```
   [root@n000] cscli support_bundle -c -t 90
   Collecting support bundle: id:4, nodes:all, time-window:90 minutes
   ```

4. Check the status of the data collection. The system response ending "in progress" indicates the export cannot occur yet.

   ```
   [root@n000] cscli support_bundle -e 22
   Support_bundle: Error: Collection of support bundle with id 22 is in progress
   ```

5. Export the support bundle.

   ```
   [root@n000] cscli support_bundle -e 22
   Support_bundle with id 22 saved in file
   support_bundle_2013-08-08_10-54-07_310920.tgz
   ```

16.3 Import a Support File

About this task

Use the Import feature to upload a single support file bundle into CSSM to view its contents.

Procedure

1. Click the CSSM **Support** tab.

   The **Support Files** screen displays.

2. Click the **Import support file** button.

   The **Import support file** dialog window opens.

3. Select a support file to upload.

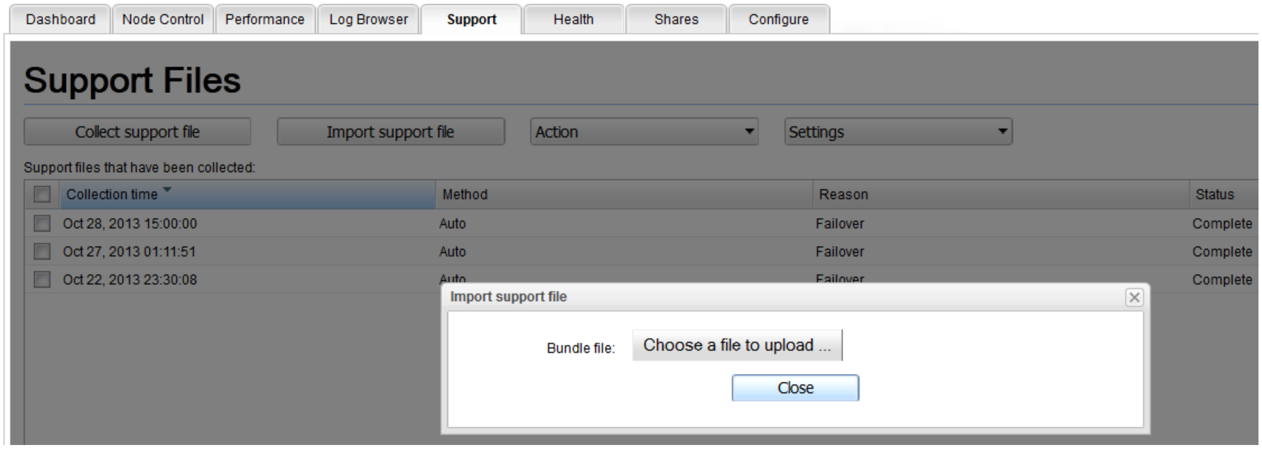
   a. Click the **Choose a file to upload ...** button.

      A list of available support files displays.

   b. Select the support file.

      The selected file opens and can be viewed.

Figure 35. Support Files Screen: Select File



Support files that have been collected:

□	Collection time	Method	Reason	Status
□	Oct 28, 2013 15:00:00	Auto	Failover	Complete
□	Oct 27, 2013 01:11:51	Auto	Failover	Complete
□	Oct 22, 2013 23:30:08	Auto	Failover	Complete

16.4 Download a Support File

About this task

Use the **Download a Copy** feature to save a local copy of the selected support file.

Procedure

1. Click the CSSM **Support** tab.

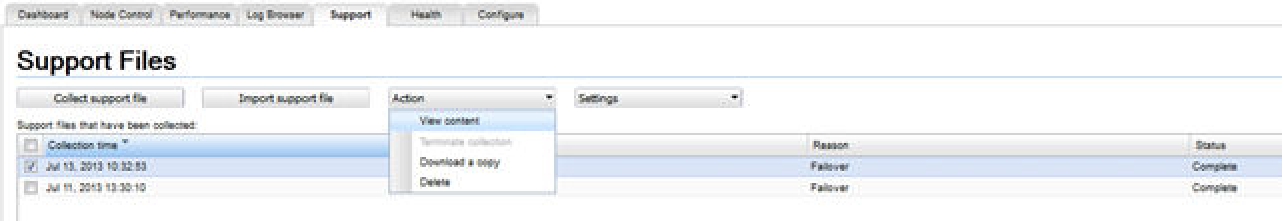
   The **Support Files** screen displays.

2. Select a support file to view.

   a. In the row containing the support file to view (in the **User actions** column) click the **Actions** button.

   b. Select **Download a copy**.

   c. Specify where to save the file on the system, or choose to open and view the file directly.



Support files that have been collected:

□	Collection time	Reason	Status
☑	Jul 13, 2013 10:32:53	Failover	Complete
□	Jul 11, 2013 13:30:10	Failover	Complete

16.5 Delete a Support File

About this task

Use the **Delete File** feature on the CSSM graphic interface to delete a selected support file:

Procedure

1. Click the CSSM **Support** tab.

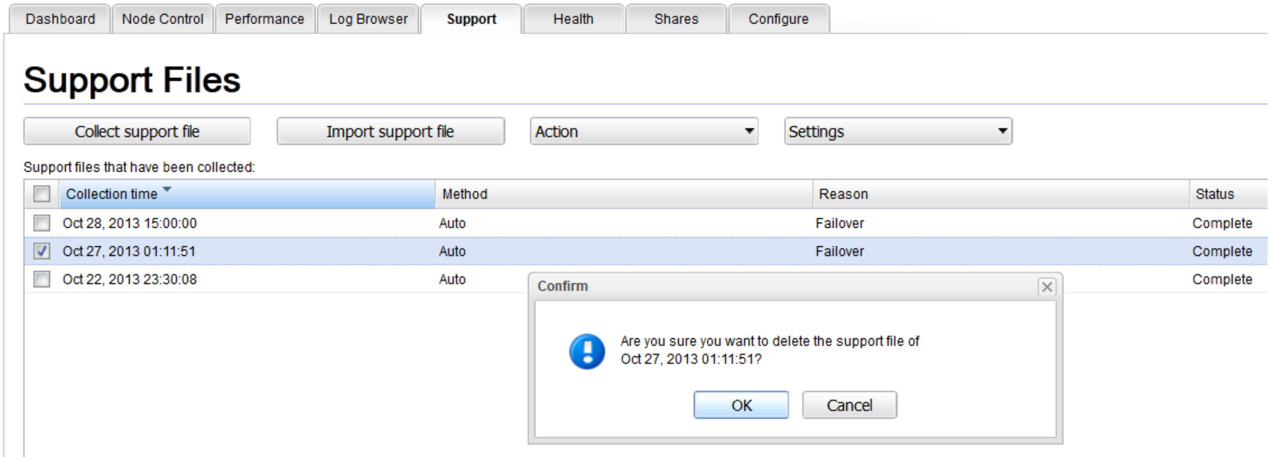
   The **Support Files** screen displays.

2. Click to select a support file.

   a. In the row containing the support file to delete (in the **User actions** column) click the **Actions** button.

   b. Select **Delete**.

   A dialog window appears prompting to confirm the deletion. Click the **Yes** button to delete the file.



Support Files

Support files that have been collected:

□	Collection time	Method	Reason	Status
□	Oct 28, 2013 15:00:00	Auto	Failover	Complete
☑	Oct 27, 2013 01:11:51	Auto	Failover	Complete
□	Oct 22, 2013 23:30:08	Auto		Complete

Confirm

Are you sure you want to delete the support file of Oct 27, 2013 01:11:51?

16.6 View Support Files

About this task

Use the **Support** tab to view the contents of a support file created either automatically or from manual data collection methods.

To view Sonexion support files:

Procedure

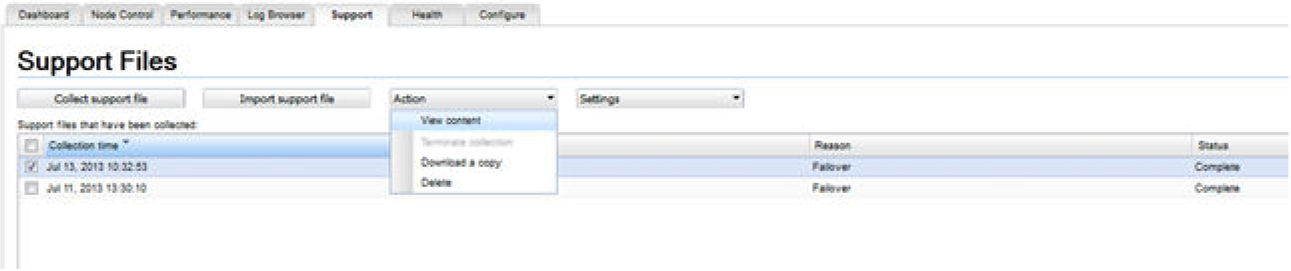
1. Click the CCMS **Support** tab.

   The **Support Files** screen is displayed, listing support files that have been collected by either the automatic or manual data collection process.

2. Select a support file to view.

a. In the row containing the support file to view (in the **User actions** column), click the **Actions** button.

b. Select **View Content**.



☐ Collection time	Reason	Status
☑ Jul 13, 2013 10:32:53	Failover	Complete
☐ Jul 11, 2013 13:30:10	Failover	Complete

The **Support File** content screen opens and displays the following tabs:

- **System Logs** (default tab) **-** Lists all the system logs for the cluster when the support file was created.
- **Node Information -** Lists information for all nodes in the cluster.
- **Web Logs -** Lists all web logs for the cluster when the support file was created.
- **Application State -** Shows data tracking the states of the management application, which is being transmitted to the support staff. Do not attempt to use this information, as it may change format from version to version.

16.7 Interpret Support Bundles

Support bundles contain two types of logs: system-wide logs that collect data for the entire system, and node-specific logs that collect data for an individual node. With the exception of lbug_syslog.csv, these log files are not intended for use by Sonexion end users, but may be valuable to Cray personnel and OEMs to better understand system states and behavior.

System-wide Logs

Log File Name	File Overview
`lbug_syslog.csv`	Contains syslog messages, in comma-separated value (CSV) format.
`logs/access.log`	Contains Apache HTTP access data.
`logs/data_tables.sql`	Contains a dump of MySQL database tables. The tables describe internal structures used to manage the cluster, the state of cluster resources, information about hardware, software, firmware, and network configuration, a FRU inventory, etc. The database dump contains all information required to recreate the system state at the time when the support bundle was created.
`logs/error.log`	Contains the Apache error log.
`logs/wsgi_access.log`	Contains records of web service calls made from the CSSM.

Node-specific Logs

Log File Name	File Overview
nodes/*nodename*/conman.log	Contains console data captured by CONsole MANager (Conman), a daemon that provides centralized access to node SOL (serial over LAN, IPMI) or real serial consoles. It also provides logging, broadcasting to several consoles or shared console sessions.
nodes/*nodename*/crm.log	Contains state data for the RAID and Lustre resources as seen by Pacemaker, an open-source, high-availability resource manager that is suitable for small and large clusters.
nodes/*nodename*/dmesg.log	Contains a dump of kernel messages collected from the node.
nodes/*nodename*/fru_dump.yaml	Contains an inventory of FRUs for the enclosure hosting the node (DDICs, PSU, fans, power supplies, etc). The dump file includes serial numbers for individual FRU equipment, firmware versions, and states such as OK or Failure).
nodes/*nodename*/lspci.log	Contains a list of PCI devices in a free-form text format generated by the lspci tool. lspci lists PCI devices and their characteristics. lspci can be run in standard or verbose (–vvv option) mode.
nodes/*nodename*/mdstat.log	Contains state data of the MDRAID arrays, i.e., content of the /proc/mdstat file.
nodes/*nodename*/processes.csv	Contains a list of processes, a snapshot of top, which is a standard monitoring program that reports the top consumers of CPU or memory.
nodes/*nodename*/sgmap.log	Contains a list of sg devices and specifies for each device the SCSI address, firmware version, and corresponding block devices.
nodes/*nodename*/software_versions	Contains a list of all installed packages with version information (rpm –qa output).
nodes/*nodename*/states.csv	Contains miscellaneous state data, including power, memory, uptime, CPU load, and Lustre targets.

17 CSCLI Reference (Release 2.1.0)

CLI commands are organized by mode; certain commands are available depending on the mode (state) of the Sonexion system. Two modes are relevant to customers: Site Configuration Mode and Daily Mode. A third mode, OEM Mode, is relevant only to Manufacturing and factory personnel. OEM Mode commands are not included in this document.

Site Configuration Mode/CustWizard Mode/Pre-Shipment Mode

Site Configuration (`custWizard`) Mode (also called "Pre-Shipment" mode) commands are available after the Sonexion system has been fully provisioned and before the system runs in Daily Mode. These commands enable users to specify customer configuration settings, apply or reset network cluster settings, obtain FRU information, upgrade Sonexion software on Lustre nodes, and toggle between Site Configuration and Daily Modes.

Daily Mode

Daily Mode commands are available after the Sonexion system has been fully provisioned and configured for customer use. These commands enable users to fully manage the Lustre file system and cluster nodes, including mount/unmount, power-cycle, failover/failback, and control node filters and exports. Daily Mode commands also enable users to obtain FRU information and upgrade Sonexion software on Lustre nodes.

Guest Mode

A guest account allows non-privileged users to run some commands to obtain information about the system, using read-only access to the system. Depending on the privileges, a subset of CSCLI commands is provided for a guest account.

Set the Mode

To determine which mode is active, run:

```
cscli cluster_mode -s
```

To change modes, run one of the following:

```
cscli cluster_mode --mode daily
cscli cluster_mode --mode custwiz
```

How CSCLI Handles Invalid Parameters

If CSCLI detects multiple invalid parameters, it may report an error for only one of them. After fixing the designated error and re-entering the command, an error is reported for the next invalid parameter, and so on.

For example, if there is a sequence of validation, when the validation of the first argument fails, the validation of upcoming arguments stops and an exception is raised.

IMPORTANT: For all CSCLI commands, exercise caution before using the `-y` or `--force` parameters, even if they are required.

Summary of Changes for Release 2.1.0

Please note:

- The CNG feature is not supported in Release 2.1.0. To avoid confusion, the section on CNG-specific CSCLI commands has been removed from this document.
- In Release 2.1.0, optional Additional DNE Units (ADUs) are now referred to as optional additional Metadata Management Units (MMUs). CSCLI commands that were previously used for ADUs are applicable to MMUs.

DailyMode - CSCLI Commands Added in Release 2.1.0

The following commands were added, changed, or removed in this software release 2.1.0:

Addition/Change	CSCLI Command	Description	Component
Added	network ean secondary	EAN Secondary interface configuration	Network
Added	network ean route	EAN routing configuration	Network
Added	ssh_port	Specify SSH TCP port settings	Network
Added	remove_unit	Remove unit(s) from cluster	Node Control

Summary of Changes for Release 2.0.0

The following table shows commands that were added or changed in the previous release:

Addition/ Change	CLI Command	Description	Component
Added	async_journal	Enables/Queries/Disables asynchronous journal for OST targets	OST Target
Added	configure_mds	Configures MDS nodes (as part of ADU addition)	ADU
Updated	service_console	Configure service console	Serviceability
Updated	support_bundle	Additional log files are available in the support bundle for MDRAID examine output and Lustre users/groups or local users/groups for CNG nodes.	Support Bundle Settings
Added	lustre_network	Lustre network (HSN) IP addresses management.	Lustre
Added	list_ranges	List ranges of IPs for HSN	Lustre
Added	find_gaps	Show unused IPs in ranges of HSN	Lustre
Added	add_range	Add new range of IPs into HSN	Lustre
Added	extend_range	Extend range of IPs for HSN	Lustre

Addition/ Change	CLI Command	Description	Component
Added	remove_range	Remove range of IPs for HSN	Lustre
Added	lustre_perf	These commands are used to view the Lustre performance	Lustre
Added	show_new_nodes	Displays the nodes in ADUs when they are added to a system	ADU
Added to existing CSCLI command	configure_hosts	Nodes in the ADU (MDS nodes) can be configured via this command	ADU
Added to existing CSCLI command	show_nodes	Nodes in the ADU (MDS nodes) will also display along with OSS nodes	ADU
Added to existing CSCLI command	fs_info	File System information of nodes (MDS) in the ADU will also display	ADU
Added	rack	Extends the functionality of cscli get_rack_position and set_rack_position	Rack
Added	UD Commands	Unresponsive Drives Commands	Disk Drive
Added	dm	DM Service Configuration Management	Disk Drive
Added	dwd	DWD - Daemon Configuration Management	Disk Drive

17.1 CSCLI Command Summary (Release 2.1.0)

The following table summarizes the 2.1.0 CSCLI command set, with columns indicating the mode or modes that include each command.

Table 6. CSCLI Command Summary

Command	Description	Daily Mode	Site Con-fig Mode	Guest Mode
`alerts`	Displays current or historical system health alert notifications	X	X	X
	`alerts elements`	X	X	X
	`alerts elements_active`	X	X	X
	`alerts nodes`	X	X	X

Command	Description	Daily Mode	Site Con-fig Mode	Guest Mode
	`alerts nodes_active`	X	X	X
	`alerts threshold`	X	X	X
`alerts_config`	Shows and updates the alerts configuration	X	X	
	`alerts_config email_add`	X	X	
	`alerts_config email_delete`	X	X	
	`alerts_config email_off`	X	X	
	`alerts_config email_on`	X	X	
	`alerts_config email_server`	X	X	
	`alerts_config email_server_update`	X	X	
	`alerts_config email_update`	X	X	
	`alerts_config thresholds`	X	X	
`alerts_notify`	Enables or disables alerts notifications	X	X	
	`alerts_notify off`	X	X	
	`alerts_notify on`	X	X	
`apply_network_setup`	Applies a network setup to a system	X	X	
`async_journal`	Enables, queries, and disables asynchronous journal for OST targets	X		
`autodiscovery_mode`	Enables or disables auto-discovery mode on system nodes	X	X	
`batch`	Runs a sequence of CSCLI commands in a batch file	X	X	
`cluster_mode`	Toggles the system between Daily mode and CustWizard (Site Configuration) mode	X	X	
`configure_hosts`	Configures host names for discovered nodes	X		
`configure_mds`	Configures MDS nodes (as part of ADU addition)	X		
`configure_oss`	Configures a new OSS node	X		
`create_filter`	Creates customer filters for nodes	X		
`delete_filter`	Deletes customer filters for nodes	X		
`dm`	*Deprecated with 2.1.0.* DM service configuration management	X		
`dwd`	*Deprecated with 2.1.0.* DWD daemon configuration management	X		

Command	Description	Daily Mode	Site Con-fig Mode	Guest Mode
`failback`	Fails back resources for the specified node	X		
`failover`	Fails over resources to the specified node	X		
`fru`	Retrieves FRU (replacement) information	X	X	X
`fs_info`	Retrieves file system information	X	X	X
`get_rack_position`	Indicates the specified node's position in the system rack	X		
`ibstat_check`	Disables or enables HA's probing of the high speed network	X		
`ip_routing`	Manages IP routing	X	X	
`list`	Lists all supported commands	X	X	X
`lustre`	Lustre management	X		X
`lustre_network`	Lustre network (HSN) IP addresses management commands	X	X	X
	`lustre_network add_range`	X	X	X
	`lustre_network defaults`	X	X	X
	`lustre_network extend_range`	X	X	X
	`lustre_network find_gaps`	X	X	X
	`lustre_network list_ranges`	X	X	X
	`lustre_network remove_range`	X	X	X
`lustre_perf`	Export lustre performance data collected by the Sonexion system	X		
`lustre_users`	Configures Lustre filesystem users	X		
	`lustre_users show`	X		
	`lustre_users local`	X		
	`lustre_users ldap`	X		
	`lustre_users ad`	X		
	`lustre_users nis`	X		
	`lustre_users upcall`	X		
	`lustre_users order`	X		
	`lustre_users apply`	X		
`manage_guest`	Manages the guest account	X		

Command	Description	Daily Mode	Site Config Mode	Guest Mode
`monitor`	Monitors the current health of the cluster nodes and elements	X	X	X
	`monitor elements`	X	X	X
	`monitor nodes`	X	X	X
`mount`	Mounts the Lustre file system in the cluster	X	X	
`netfilter_level`	Manages the netfilter level	X	X	
`network`	Appliance network configuration	X		X
	`apply`	X		X
	`ean`	X		X
	`show`	X		X
`power_manage`	Specifies node power management options	X		
`rack`	Used to list, show, create, delete, move and rename racks.	X		
`raid`	RAID management	X		X
	`raid check`	X		X
	`raid disk_fail`	X		X
	`raid speed`	X		X
	`raid show`	X		X
`raid_check`	*Deprecated with 2.1.0.* Enables RAID checks on RAID devices. See the `raid` command.	X		
`rebuild_rate`	*Deprecated with 2.1.0.* Manages the RAID rebuild rate. See the `raid` command.	X	X	
`remove_unit`	Remove units from cluster.	X		
`reset_network_setup`	Resets the network setup of an existing Sonexion system		X	
`restore_mgmt`	Enables MGMT node recovery. When enabled, MGMT node that boots is restored from the latest good backup.	X		
`service_console`	Configure service console	X		
`set_admin_passwd`	Changes administrator user password on an existing Sonexion system	X	X	
`set_date`	Manages the date setting on the system	X		
`set_network`	Specifies the system network setup		X	
`set_node_version`	Changes the image for a diskless node	X		

Command	Description	Daily Mode	Site Con-fig Mode	Guest Mode
set_rack_position	Changes a given node position in a system rack	X		
set_timezone	Manages the timezone setting on the system	X		
show_filters	Shows customized and predefined node filters	X		X
show_network_setup	Shows the system's network setup		X	X
show_new_nodes	Displays a table with new OSS nodes and their resources	X		
show_node_versions	Shows the current software version on the specified nodes	X		
show_nodes	Displays node information	X		X
show_update_versions	Shows available software versions in the Sonexion Management Server repository	X		
show_version_nodes	Shows all nodes at the specified software version	X		
sm	Manages the InfiniBand Subnet Manager	X	X	
ssh_port	Specify SSH TCP port settings.	X	X	
ssl	Manage the SSL certificate.	X		X
support_bundle	Manages support bundles and support bundle settings	X	X	X
syslog	Retrieves syslog entries	X	X	X
unmount	Unmounts Lustre clients or targets on the file system	X	X	
update_node	Updates the software version on the specified node	X		

17.2 Administrative Commands

Administrative commands include functions to get file system and cluster node information, retrieve syslog entries, show FRU information and list available commands.

fs_info File System Information Command

Modes: Site configuration, Guest, Daily

The fs_info command shows all file system information. File System information of nodes (MDS) will also display if they are configured.

Synopsis

```
$ cscli fs_info [-h] [-f fs_name] |-c cluster_name |--cluster cluster_name
```

where:

Optional Arguments	Description
-h |--help	Displays the help message and exits.
-f *fs_name* |--fs *fs_name*	Displays the file system name.
-c *cluster_name*|--cluster *cluster_name*	*This parameter is deprecated.* It is supported only for backward compatibility.

`fru` Information Command

Modes: Site configuration, Guest, Daily

The `fru` command retrieves information about Field-Replaceable Units (FRUs) in the Sonexion system. FRUs are grouped into the following element 'types': ArrayDevice, BMC, Cooling, Enclosure, Enclosure_Electronics, PSU and Battery. FRU information can be retrieved per element type, on a per-node basis, or for all nodes in the system.

Synopsis

```
$ cscli fru [-h] (-a | -n node_spec) [-t
{ArrayDevice,BMC,Cooling,Enclosure,Enclosure_Electronics,PSU,Battery}] [-i index]
[-l [history]]
```

where

Optional Arguments	Description
-h |--help	Displays the help message and exits.
-a|--all	Displays FRUs (including status) grouped by type, for all nodes in the system.
-n *node_spec*|--nodes *node_spec*	Displays FRUs (including status) grouped by element type, for a specified node(s) in the system.
-t {ArrayDevice, *bmc*, Cooling, Enclosure, Enclosure_Electronics, PSU, Battery}	Displays FRUs (including status) for the specified element type. Examples of element types: array device, BMC, PSU, battery.
-i *index* |--index *index*	Displays FRUs (including status) for specified elements within a list of elements of the same type.
-l [*history*] | --history [*history*]	Displays FRU history (default is 10 lines of history).

`cluster_mode` Command

Modes: Site configuration, Daily

The `cluster_mode` command toggles the system among multiple system modes: daily, customer Wizard, or pre-shipment.

Synopsis

```
$ cscli cluster_mode [-h] [-s] [--mode {daily,custwiz,pre-shipment}] [--db-only]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-s |--status`	Displays the status of the cluster.
`--mode {daily,custwiz,pre-shipment}`	Switches to the specified mode. Switches to daily mode, site configuration mode or pre-shipment mode.
`--db-only`	Update only the database. Does not sync nodes via puppet. Valid only with '--mode' argument.

`list` Commands in Mode Command

Modes: Site configuration, Guest, Daily

The `list` command shows a list of available commands in the current Sonexion mode.

Synopsis

```
$ cscli list [-h]
```

where:

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

`syslog` Display Log Information Command

Modes: Site configuration, Guest, Daily

The `syslog` command displays Lustre log entries.

Synopsis

```
$ cscli syslog [-h] [-m max] [-F] [-d duration] [-s start_time] [-e end_time] [-r]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-m` *max* `|--max=`*max*	Specifies the maximum number of entries to return.
`-F |--follow`	Polls for future messages. Only valid without `-e`, `-r` arguments.
`-d` *duration* `|--duration=`*duration*	Specifies duration (in seconds) for which to follow output. Only valid with `-F` argument.
`-s` *start_time* `|--start_time=`*start_time*	Specifies the earliest time for which messages should be received.
`-e` *end_time* `|--end_time` *end_time*	Specifies the latest time for which messages should be received.
`-r |--reverse`	Sorts entries in descending order (by time).

`set_admin_passwd` Command

Modes: Site configuration, Daily

The `set_admin_passwd` command changes and sets an administrator password.

Synopsis

```
$ cscli set_admin_passwd [-h] [-p password]
```

Optional Arguments	Description
`-h|--help`	Displays the help message and exits.
`-p|--password`	Specify the new administrator password string.

`batch` Command

Modes: Site configuration, Daily

The `batch` command runs a sequence of CSCLI commands in a batch file.

Synopsis

```
$ cscli batch [-h] –b batch_file
```

where:

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-b batch_file|--batch-file batch_file`	Specifies the command batch file.

`ip_routing` Command

Modes: Site configuration, Daily

The `ip_routing` command manages IP routing to and from the system database.

Synopsis

```
$ cscli ip_routing [arguments]
```

where arguments can be **one** of the following lines:

```
--show|-s [--loadable]
```

```
 --load path_to_file
```

```
 --add | -a  --dest destination_ip --prefix prefix_len --router router_ip
```

```
 --update | -u --route-id route_id [--dest destination_ip] [--prefix prefix_len]
[--router router_ip]
```

```
 --delete | -d --route-id route_id
```

```
 --clear | -c
```

```
 --apply | -a
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-s |--show`	Shows the current IP routing table in the database.
`--loadable`	Prints the routing table in loadable format (use with the `–show` argument).
`-c |--clear`	Clears the routing table in the database.
`--apply`	Applies IP routing.
`--load` *`load`*	Loads the IP routing table from a file to the database.
`-a |--add`	Inserts IP routing in the database.
`-u |--update`	Updates IP routing in the database.
`-d |--delete`	Deletes IP routing from the database.
`--dest` *`dest`*	Specifies the destination IP address.
`--prefix` *`prefix`*	Specifies the prefix length (0-32).
`--router` *`router`*	Specifies the router IP address.
`--route-id` *`route_id`*	Specifies the route identifier (see `ip_routing –show`).

`ssl` Command

The `ssl` command is used to manage the SSL certificate.

Synopsis

```
$ cscli ssl [-h] {install,show}
```

where:

Positional Arguments	Description
`show`	Show currently installed certificate.
`install`	Install new SSL certificate.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

`ssl install` Subcommand

The `ssl install` command is a subcommand of the `ssl` command, and is used to install a new SSL certificate.

Synopsis

```
$ cscli ssl install [-h] -f FILE
```

where:

Optional Arguments	Description

`-f FILE`
`|--cert-file FILE`
	PEM file containing the new certificate and key.

`-h |--help`	Displays the help message and exits.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

17.3 Administrative Settings Commands

Manage the Administrative Password

Modes: Site configuration, Guest, Daily

The `set_admin_passwd` command sets the Sonexion system administrator's user password.

Synopsis

```
$ cscli set_admin_passwd [-h] -p password
```

Option	Description
`-h |--help`	Prints the help message and exits.
`-p |--password`	Sets the system administrator's password.

Manage the System Date

The `set_date` command manages the system date.

IMPORTANT: Exercise caution before using the `--force-ntp` parameter.

Synopsis

```
$ cscli set_date [-h] [-s new_date] [--force-ntp]
```

Optional Arguments

Option	Description
`-h |--help`	Displays the help message and exits.
`-s` *`new_date`* `|--set` *`new_date`*	Specifies the new date in the format: *`MMDDhhmmCCYY.ss`*
`--force-ntp`	*Deprecated in 3.0.0*. Forces NTP configuration.

Manage the System Timezone

Modes: Daily

The `set_timezone` command manages the system time zone setting.

Synopsis

```
$ cscli set_timezone [-h] [-s new_timezone] [-l]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-s new_timezone |--set new_timezone`	Specifies the new time zone location name. For example, "America/Los_Angeles".
`-l |--list`	Lists the available time zones.

Changing the time zone is optional. By default, the time zone is set to PDT (UTC/GMT -7). If necessary, the time zone setting can be changed. However after using the `cscli set_timezone` command on the system, it is not propagated completely to all components. Rebooting is the safest way to ensure that all processes are using the same time zone. Thus it is recommended to do a full system restart of the system after setting the time zone.

Manage the InfiniBand Subnet Manager

Modes: Site configuration, Guest, Daily

The `sm` command manages (enables, disables or prioritizes) the InfiniBand Subnet Manager (SM) integrated with the Sonexion system. The local SM ensures that InfiniBand is properly configured and enabled for use. In situations in which the system is connected to a larger InfiniBand network that already uses a subnet manager, the local SM should be disabled. The `sm` command can also be used to modify subnet manager priorities.

Synopsis

```
$ cscli sm [-h] (-e | -d) [-P {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15}]|-c
cluster_name |--cluster cluster_name
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-e |--enable`	Enables the IB storage manager used with the system.
`-d |--disable`	Disables the IB storage manager used with the system.
`-P {0,1,2,3,4,5,6,7,8,9,10,11,12,13, 14,15}|--priority {0,1,2,3,4, 5,6,7,8,9,10, 11,12,13,14,15}`	Sets the priority [0..15] of the IB storage manager used with the system.
`-c cluster_name|--cluster cluster_name`	*This parameter is deprecated.* It is supported only for backward compatibility.

Guest Account Management

Modes: Daily

A guest account allows a non-privileged user to run some commands to obtain information about the system (read-only access to the appliance). Depending on the privileges CSCLI provides a limited subset of commands to the “guest” account.

The “Guest” account has access to export but that displays only the subcommands.

The `cscli manage_guest` command is part of the "user management" component and is available for an "admin" account only. This command lets you enable or disable a guest account access using UI or/and CSCLI and change the guest account password.

Synopsis

```
$ cscli manage_guest [-h] [-s] [--enable-shell | --disable-shell]
[--enable-web | --disable-web] [--password password | --password-disable]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-s |--show`	Displays guest account information.
`--enable-shell`	Enables shell for guest account.
`--disable-shell`	Disables shell for guest account.
`--enable-web`	Enables web for guest account.
`--disable-web`	Disables web for guest account.
`--password` *`password`*	Guest account password.
`--password-disable`	Clears password and blocks account until a new password is set. This option also sets “disabled” flags in cscli and GUI for guest account.

17.4 `alerts` Display Command

Modes: Site configuration, Guest, Daily

The `alerts` command displays current and historic health alerts for system nodes and elements, and thresholds for system alerts.

Synopsis

```
$ cscli alerts [-h] {elements_active,nodes,elements,nodes_active,thresholds}
```

Positional Arguments	Description
`nodes`	Displays alert history for nodes.
`elements`	Displays alert history for elements.
`nodes_active`	Displays current alerts for nodes.
`elements_active`	Displays current alerts for elements.

Positional Arguments	Description
thresholds	Displays editable alert thresholds and their current settings.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

alerts elements_active Subcommand

Synopsis

```
$ cscli alerts elements_active [-h] [-y] [-v] [-x] [-n node_spec | -g
genders_query] [-S element_filter]
```

Optional Arguments	Description
-h |--help	Displays the help message and exits.
-y |--yaml	Outputs data in YAML format.
-v |--verbose	Outputs extra data.
-x |--unhandled	Displays alerts for notifications that have not been turned off. (Default value is all alerts are shown.)
-n *node_spec* |--nodes *node_spec*	Specifies pdsh-style node hostnames (e.g. node[100-110,120]).
-g *genders_query* |--genders *genders_query*	Specifies node genders attributes query (e.g. mds=primary).
-S *element_filter* |--search *element_filter*	Specifies the element filter so a search can be done by element name. The pattern is case-sensitive. Regular expressions allowed. For example, Fan Statistics, Power Statistics, Thermal Statistics, Voltage Statistics, etc.

alerts nodes Subcommand

Synopsis

```
$ cscli alerts nodes [-h] [-y] [-s start_time] [-e end_time] [-m limit] [-n
node_name] [-N {down,unreachable,up}]
```

Optional Arguments	Description
-h |--help	Displays the help message and exits.
-y |--yaml	Outputs data in YAML format.
-s *start_time* |--start-time *start_time*	Specifies the alert start time in ISO-8601 format. If --start-time is not specified, then --end-time is ignored and the "last 7 days" period is used.

Optional Arguments	Description
-e *end_time* |--end-time *end_time*	Specifies the alert end time in ISO-8601 format. (Default value is “now”.)
-m *limit* |--max *limit*	Specifies the maximum number (limit) of alerts to display.
-n *node_name* |--*node node_name*	Specifies the node for which to display alerts. **Pdsh-**style node masks are not allowed here.
-N {down,unreachable,up}|--node *status*	Specifies node status.

alerts elements Subcommand

Synopsis

```
$ cscli alerts elements [-h] [-y] [-s start_time] [-e end_time] [-m limit] [-n
node_name] [-U {unknown,warning,ok,critical}]
```

Optional Arguments	Description
-h |--help	Displays the help message and exits.
-y |--yaml	Outputs data in YAML format.
-s *start_time* |--start-time *start_time*	Specifies the start time filter in ISO-8601 format. If -–start-time is not specified, --end-time is ignored and the "last 7 days" period is used.
-e *end_time* |--end-time *end_time*	Specifies the end time filter in ISO-8601 format (default value is "now").
-m *limit* |--max *limit*	Specifies the maximum number (limit) of items to display.
-n *node_name* |--node *node_name*	Specifies the node for which to display items. Pdsh-style node masks are not allowed here.
-U {unknown,warning,ok,critical}| --element *status*	Specifies the element's status.

alerts nodes_active Subcommand

Synopsis

```
$ cscli alerts nodes_active [-h] [-y] [-v] [-x] [-n node_spec | -g genders_query]
```

Optional Arguments	Description
-h |--help	Displays the help message and exits.
-y |--yaml	Outputs data in YAML format.
-v |--verbose	Outputs extra data.

Optional Arguments	Description
`-x |--unhandled`	Displays alerts for notifications that have not been turned off (default is all alerts are shown).
`-n` *`node_spec`* `|--nodes` *`node_spec`*	Specifies pdsh-style node hostnames (e.g. node[100-110,120]).
`-g` *`genders_query`* `|--genders` *`genders_query`*	Specifies node genders attributes query (e.g. `mds=primary`).

`alerts threshold` Subcommand

Synopsis

```
$ cscli alerts thresholds [-h] [-y]
```

Threshold fields:

Field	Description
name	Short identifier of the threshold
`description`	Describes the threshold and gives tips on how to modify it
`gender`	Type of nodes to which the threshold is applied
`warning`	Value of the warning threshold
`critical`	Value of the critical threshold

Possible gender values:

Value	Specifies
all	All nodes; general node type that can be overwritten by more specific node types
`mgmt`	Management nodes (primary and secondary)
`mds`	Metadata Servers
`oss`	Object Storage Servers

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-y |--yaml`	Outputs data in YAML format.

17.5 `alerts_config` Command

Modes: Site configuration, Daily

The alerts_config command allow viewing and updating of the alerts configuration.

Synopsis

```
$ cscli alerts_config [-h] {email_off,thresholds,email_update,
email_server_update, email_delete, email_add,email_on,email_server,emails}
```

Positional Arguments	Description
email_off	Turns off notifications for notification subscribers.
thresholds	Sets the current value of an threshold. This value can be edited.
email_update	Sends an email alert with an update.
email_server_update	Sends an email alert with a server update.
email_delete	Deletes the email.
email_add	Adds a new notification subscriber.
email_on	Turns on notifications for subscribers.
email_server	Displays the relay SMTP server configuration.
emails	Lists the alert notification subscribers.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

email_off Subcommand

The email_off command turns off notifications for subscribers.

Synopsis

```
$ cscli alerts_config email_off [-h] -u email
```

Optional Arguments	Description
-h |--help	Displays the help message and exits.
-u email |--user email	Displays subscriber email. Notifies subscribers have new mail in /var/spool/mail or admin.

thresholds Subcommand

Current thresholds are applied to the monitoring configuration only if the --apply-config option is used. It may take about 15 seconds to apply the configuration threshold changes.

If a group of changes needs to be made to the thresholds, edit a few threshold values and then add the --apply-config option to the last edit to set all the changes at once.

The new thresholds applied to monitoring configuration take effect a few minutes after they are applied when the next scheduled node check is performed.

The only editable thresholds are those listed in the output of the cscli alerts thresholds command.

Synopsis

```
$ cscli alerts_config thresholds [-h] -t threshold_name -g gender_name [-W
warning_threshold_value] [-C critical_threshold_value] [-A]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-t` *`threshold_name`* `|--threshold` *`threshold_name`*	Displays the name of the threshold.
`-g` *`gender_name`* `|--gender` *`gender_name`*	Displays the gender name of the threshold.
`-W` *`warning_threshold_value`* `|--warning` *`warning_threshold_value`*	Displays the warning threshold value.
`-C` *`critical_threshold_value`* `|--critical` *`critical_threshold_value`*	Displays the critical threshold value.
`-A |--apply-config`	Applies the threshold configuration.

If the `--apply-config` command is used the current thresholds are applied only to the monitoring configuration. It may take about 15 seconds to apply the configuration threshold changes.

To make a group of changes to the thresholds, edit a few threshold values and then add the `--apply-config` option with the last edit to set all the changes at once.

The new thresholds applied to monitoring configuration take effect a few minutes after they are applied when the next scheduled node check is performed.

The only editable thresholds are those listed in the output of the `cscli alerts thresholds` command.

`email_update` Subcommand

The `email_update` command updates the existing subscriber's notification.

Notification Levels

The `level` option sets the alerts trigger for an email to be sent to a subscriber. The possible level option values are:

- **Critical** - Notify elements critical or node down statuses
- **Warning** - Notify elements warning statuses
- **Unknown** - Notify elements unknown statuses
- **Ok** - Notify when elements and nodes recover from problems
- Any combination of the above (comma-separated)
- **None** - No notifications (similar to `cscli alerts_config email_off`)
- **All** - Send all notifications, including notifications
- When a node/element is flapping between statuses
- When a node/element is in scheduled downtime

When the email notification level for a subscriber is set to "OK", an email for an "OK" alert is not sent to the subscriber. For the system to send email for "OK" alerts, make sure that at least one additional severity of alerts is given.

Notification Periods

The Notification periods are as follows:

- 24x7 - Notify always
- Workhours - Notify only during working days and hours (in the timezone of the server).

Synopsis

```
$ cscli alerts_config email_update [-h] -u email [-M email] [-N user_full_name]  [-
P {24x7,workhours}] [-L level]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-u` *`email`* `|--user` *`email`*	Displays subscriber email. Notifies you have new mail in `/var/spool/mail` or `admin`.
`-M` *`email`* `|--email` *`email`*	Displays the email address.
`-N` *`user_full_name`* `|--name` *`user_full_name`*	Displays a longer name or description for the subscriber.
`-P {24x7,workhours} |--period {24x7,workhours}`	Displays the time periods at which the subscriber is notified. possible values: {24x7,workhours}
`-L` *`level`*`, |--level` *`level`*	Displays notification level; possible values: any comma-separated combination of `{critical,ok,unknown,warning}`, or `all`, or `none`.

`email_server_update` Subcommand

The `email_server_update` command configures the SMTP server to send alerts to external email addresses.

Synopsis

```
$ cscli alerts_config email_server_update [-h] -s smtp_server_address [--port
port] [-S email_from] [-d domain] [-u smtp_user] [-p smtp_password]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-s` *`smtp_server_address`* `|--server` *`smtp_server_address`*	Displays an IP address or hostname of the (relay) SMTP server.
`--port` *`port`*	SMTP server port (default: 25)
`-S` *`email_from`*`|--sender` *`email_from`*	Displays the senders email address.

Optional Arguments	Description
	If `--domain` is set, the default value for the sender is *cluster_name@domain*. If `--domain` is not set, the sender's email address is required.
`-d` *domain* `|--domain` *domain*	Displays the internet hostname of the mail system to be used with email addresses that have no "@".
`-u` *smtp_user*, `--user` *smtp_user*	Specifies the username if the SMTP server requires authentication.
`-p` *smtp_password* `|--password` *smtp_password*	The password if the SMTP server requires authentication.

email_delete Subcommand

The `email_delete` command deletes notifications to subscribers.

Synopsis

```
$ cscli alerts_config email_delete [-h] -u email
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-u` *email* `|--user` *email*	Displays subscriber email. Notifies you have new mail in `/var/spool/mail` or admin.

email_add Subcommand

The `email_add` command adds a new notification subscriber.

Synopsis

```
$ cscli alerts_config email_add [-h] -M email [-N user_full_name] [-P
{24x7,workhours}] [-L level]
```

Notification Levels

The level option sets the alerts trigger for email to be sent to a subscriber. Possible level option values are:

- **Critical** - Notify elements critical or node down statuses
- **Warning** - Notify elements warning statuses
- **Unknown** - Notify elements unknown statuses
- **Ok** - Notify when elements and nodes recover from problems
- Any combination of the above (comma-separated)
- **None** - No notifications (similar to "cscli alerts_config email_off")
- **All** - Send all notifications, including notifications when a node/element is flapping between statuses, or when a node/element is in scheduled downtime

Notification Periods

Possible Notification Periods:

- 24x7 - Notify always
- Workhours - Notify only during working days and hours (in the timezone of the server) .

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-M` *`email`*`|--email` *`email`*	Displays subscriber email. Notifies you have new mail in `/var/spool/mail` or admin. email address.
`-N` *`user_full_name`* `|--name` *`user_full_name`*	Displays a longer name or description for the subscriber.
`-P {24x7,workhours} |--period {24x7,workhours}`	The time periods at which the subscriber is notified. Possible values: `{24x7,workhours}` (default: `24x7`).
`-L` *`level`* `|--level` *`level`*	The notification level. Possible values: any comma-separated combination of: `{critical,ok,unknown,warning}`, or `all`, or `none` (default: `all`).

`email_on` Subcommand

The `email_on` command turns on notifications for subscribers.

Synopsis

```
$ cscli alerts_config email_on [-h] -u email
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-u` *`email`* `|--user` *`email`*	Displays subscriber email. Notifies you have new mail in `/var/spool/mail` or admin.

`email_server` Subcommand

The `email_server` command displays the relay smtp server configuration.

Synopsis

```
$ cscli alerts_config email_server [-h]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

`emails` Subcommand

The `emails` command displays a list of alert notifications to the subscribers.

Notification Levels

The level option sets the alerts trigger for email to be sent to a subscriber. Possible level option values are:

- **Critical** - Notify elements critical or node down statuses
- **Warning** - Notify elements warning statuses
- **Unknown** - Notify elements unknown statuses
- **Ok** - Notify when elements and nodes recover from problems
- Any combination of the above (comma-separated)
- **None** - No notifications (similar to "cscli alerts_config email_off")
- **All** - Send all notifications, including notifications
- When a node/element is flapping between statuses
- When a node/element is in scheduled downtime

Synopsis:

```
$ cscli alerts_config emails [-h] [-y] [-v] [-u email]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-y |--yaml`	Outputs data in YAML format.
`-v |--verbose`	Outputs extra data in verbose mode.
`-u` *`email`* `|--user` *`email`*	Displays subscriber email.Notifies you have new mail in `/var/spool/mail` or `admin`.

17.6 `alerts_notify` Command

Modes: Site configuration, Daily

The `alerts_notify` command turns alert notifications on or off.

Synopsis

```
$ cscli alerts_notify [-h] {on,off} ...
```

Positional Arguments	Description
`on`	Sets the alert notification on.
`off`	Sets the alert notification off.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

alerts_notify on

The alerts_notify on command turns alert notifications on.

Synopsis

```
$ cscli alerts_notify on [-h] (-n node_spec | -g genders_query) [-S element_filter
| -E element_name]
```

Positional Arguments	Description
-h |--help	Displays the help message and exits.
-n node_spec |--node |--*node_spec* |--nodes *node_spec*	Looks through passed hostname elements. Looks for pdsh style nodes host names (e.g. node[100-110,120]).
-g *genders_query*|--genders *genders_query*	Displays the node genders attributes query (e.g. mds=primary).
-S *element_filter* |--search *element_filter*	This command searches by element name. The pattern is case sensitive. Regular expressions allowed.
-E *element_name*|--element *element_name*	Displays the element name.

alerts_notify off

The alerts_notify off command turns alert notifications off.

Synopsis

```
$ cscli alerts_notify off [-h] (-n node_spec | -g genders_query) [-S
element_filter | -E element_name] [-C comment]
```

Positional Arguments	Description
-h |--help	Displays the help message and exits.
-n node_spec |--node |--*node_spec* |--nodes *node_spec*	Looks through passed hostname elements. Looks for pdsh style nodes host names (e.g. node[100-110,120]).
-g *genders_query*|--genders *genders_query*	Displays the node genders attributes query (e.g. mds=primary).
-S *element_filter* |--search *element_filter*	This command searches by element name. The pattern is case sensitive. Regular expressions allowed.
-E *element_name*|--element *element_name*	Displays element name.
-C *comment*|--comment *comment*	Displays a brief description of the action being taken.

17.7 Appliance Network Configuration Commands

The network commands are used to configure the appliance network. *(Introduced with 2.1.0.)*

Synopsis

```
$ cscli network [-h] {apply,ean,show}
```

where:

PositionalArguments	Description
show	Show networks.
apply	Apply network configuration changes.
ean	EAN (External Administration Network) configuration.

OptionalArguments	Description
-h |--help	Displays the help message and exits.

network apply Subcommand

The network apply command is a subcommand of the network command.

Synopsis

```
$ cscli network apply [-h]
```

where:

OptionalArguments	Description
-h |--help	Displays the help message and exits.

network show Subcommand

The network show command is a subcommand of the network command.

Synopsis

```
$ cscli network show [-h]
```

where:

OptionalArguments	Description
-h |--help	Displays the help message and exits.

network ean Subcommand

The ean command is a subcommand of the network command. Use it to configure system access to the External Administration Network (EAN).

Synopsis

```
$ cscli network ean [-h] {ntp,route,dns,secondary}
```

where:

Positional Arguments	Description
ntp	EAN NTP configuration.
dns	EAN DNSconfiguration.
secondary	EAN Secondary interfaceconfiguration.
route	EAN Routing configuration.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

network ean ntp Subcommand

The network ean ntp command is the second level subcommand of the networkean command. Use it to set or clear NTP servers.

Synopsis

```
$ cscli network ean ntp [-h] {clear,set}
```

where:

Positional Arguments	Description
set	Set NTPservers.
clear	Clear NTP servers.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

network ean dns Subcommand

The network ean dns command is the second level subcommand of the network ean command. Use it to set or clear DNS servers.

Synopsis

```
$ cscli network ean dns [-h] {clear,set}
```

where:

Positional Arguments	Description
set	Set DNS servers.
clear	Clear DNS servers.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

network ean secondary Subcommand

The network ean secondary command is the second level subcommand of the network ean command. Use it to manage secondary EAN interfaces.

Synopsis

```
$ cscli network ean secondary [-h] {add,show,delete}
```

where:

PositionalArguments	Description
add	Add secondary EAN interface.
show	Delete secondary EAN interface.
delete	Show list of secondary EAN interfaces.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

network ean route Subcommand

The network ean route command is the second level subcommand of the network ean command. Use this command to manage network routing.

Synopsis

```
$ cscli network ean route [-h] {load,set,show,clear,add,apply,delete}
```

where:

Positional Arguments	Description
add	Add routing rule.
delete	Delete routing rule.
set	Update routing rule.
load	Load list for routing rules.

Positional Arguments	Description
clear	Clear routing rules.
apply	Apply routing rules.
show	Show list of routes.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

network ean route add Subcommand

The network ean route add command is a second level subcommand of the network ean route command. Use it to add a network routing rule.

Synopsis

```
$ cscli network ean route add [-h] -i IFACE -d DEST -p PREFIX -g GATEWAY
```

where:

Optional Arguments	Description
-i IFACE |--iface IFACE	Interface name.
-d DEST |--dest DEST	Destination ip address.
-p PREFIX |--prefix PREFIX	Prefix length (0-32).
-g GATEWAY |--gateway GATEWAY	Gateway ip address.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

network ean route delete Subcommand

The network ean route delete command is a second level subcommand of the network ean route command. Use it to delete a network routing rule.

Synopsis

```
$ cscli network ean route delete [-h] -r ROUTE_ID
```

where:

Optional Arguments	Description

`-r ROUTE_ID`
`|--route-id ROUTE_ID`
	Route identifier(see `ean route show`).

`-h |--help`	Displays the help message and exits.

network ean route set Subcommand

The `network ean route set` command is a second level subcommand of the `network ean route` command. Use the command to update a routing rule.

Synopsis

```
$ cscli network ean route set [-h] -r ROUTE_ID [-d DEST] [-p PREFIX][-gGATEWAY]
```

where:

Optional Arguments	Description

`-r ROUTE_ID`
`|--route-id ROUTE_ID`
	Route identifier (see `ean route show`).

`-d DEST`
`|--dest DEST`
	Destination ip address.

`-p PREFIX`
`|--prefix PREFIX`
	Prefix length (0-32).

`-g GATEWAY`
`|--gateway GATEWAY`
	Gateway ip address.

`-h |--help`	Displays the help message and exits.

network ean route load Subcommand

The network ean route load command is a second level subcommand of the network ean route command. Use to load routing rules from a file.

Synopsis

```
$ cscli network ean route load [-h] -i IFACE -f FILENAME
```

where:

Optional Arguments	Description

`-i IFACE`
`|--iface IFACE`
	Interface name.

Optional Arguments	Description

`-f FILENAME`
`|--file FILENAME`
	Path to file with routing rules.

`-h |--help`	Displays the help message and exits.

`network ean route clear` Subcommand

The `network ean route clear` command is a second level subcommand of the `network ean route` command. Use the command to clear routing rules.

Synopsis

```
$ cscli network ean route clear [-h] [-i IFACE]
```

where:

Optional Arguments	Description

`-i IFACE`
`|--iface IFACE`
	Interface name.

`-h |--help`	Displays the help message and exits.

`network ean route apply` Subcommand

The `network ean route apply` command is a second level subcommand of the `network ean route` command. Use the command to apply a routing rule.

Synopsis

```
$ cscli network ean route apply [-h]
```

where:

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

`network ean route show` Subcommand

The `network ean route show` command is a second level subcommand of the `network ean route` command.

Synopsis

```
$ cscli network ean route show [-h]
```

where:

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

17.8 Asynchronous Journal - `async_journal` - Command

Modes: Daily

The `async_journal` command can be used to enable, query or disable OST targets.

Synopsis

```
$ cscli async_journal [-h] [-s] [--enable] [--disable]
```

where:

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-s |--status`	Displays the status of asynchronous journal.
`--enable`	Enables asynchronous journal for OSTs.
`--disable`	Disables asynchronous journal for OSTs.

17.9 Configuration Commands

The configuration commands specify the MAC address and hostname for a given node and configure OSS nodes.

`configure_hosts` Command

Modes: Daily

The `configure_hosts` command configures the MAC address and host names for the discovered node. Nodes in the ADU (MDS nodes) can be configured via this command

Synopsis

```
$ cscli configure_hosts [-h] -m mac_address --hostname hostname [-f]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-m` *`mac_address`* `|--mac` *`mac_address`*	The node *`mac_address`*
`--hostname` *`hostname`*	The new node hostname.
`-f |--force`	Forces the mode (to skip hostname validation).

configure_mds Command

Modes: Daily

The configure_mds command is used to add and configure new MDS nodes (optional additional MMUs) in the Sonexion system. The command is used in two modes:

- Bind MD device with Lustre FS (by name)
- Apply all bindings (configuration), i.e., perform formatting of MD devices as Lustre targets, configuration of HA, and so forth.

To bind MD device with Lustre FS (by name), the following options can be used, for example:

```
# cscli configure_mds -c snx11000n -n snx11000n06 --bind-arrays md0:snx11000n
# cscli configure_mds -c snx11000n -n snx11000n07 --bind-arrays md1:snx11000n
```

To apply all bindings (configuration), the following option can be used, for example:

```
# cscli configure_mds -c snx11000n -n snx11000n06 --apply-config
```

Synopsis

```
$ cscli configure_mds [-h] -n node_spec (-A | -b bind_arrays)
|-c cluster_name |--cluster cluster_name
```

where:

Optional Arguments	Description
-h | --help	Displays the help message and exits.
-n node_spec | --nodes *node_spec*	Specifies the hostname of the new MDS node (in genders style).
-a | --apply-config	Applies the configuration to the new MDS node.
-b *bind_arrays* | --bind-arrays *bind_arrays*	Specifies comma-separated pairs of array-file system bindings. Each binding should be in this format: *array*:*file_system_name*. The *array* variable can be a genders-style string such as **md[0-3]**.
-c *cluster_name* | --cluster *cluster_name*	*This parameter is deprecated.* It is supported only for backward compatibility.

configure_oss Command

Modes: Daily

The configure_oss command configures new OSS nodes in the Sonexion system.

Synopsis

```
$ cscli configure_oss [-h] -n node_spec (-A | -b bind_arrays)
|-c cluster_name |--cluster cluster_name
```

Optional Arguments	Description
-h |--help	Displays the help message and exits.

Optional Arguments	Description
`-n` *`node_spec`* `|--nodes` *`node_spec`*	Specifies the hostname of the new OSS node (in genders style).
`-A |--apply-config`	Applies the configuration to the new OSS node.
`-b bind_arrays |--bind-arrays` *`bind_arrays`*	Specifies comma-separated pairs of array-file system bindings. Each binding should be in this format:array:file_system_name. The array variable can be a genders-style string, e.g. `md[0-3]`.
`-c` *`cluster_name`*`| --cluster` *`cluster_name`*	*This parameter is deprecated.* It is supported only for backward compatibility.

`restore_mgmt` Command

Modes: Daily

The `restore_mgmt` command enables MGMT node recovery. When enabled, the MGMT node that boots is restored from the latest good backed-up nodes. This command is used to format and copy data to the internal drives on the MGMT nodes.

When run on MGMT0 node with the `--enable` parameter, it copies data from the MGMT node backup image (which is created using a nightly cron job) to the internal drive on the MGMT1 node. When run with the `--enable` command on the MGMT1 node, it will do likewise on the MGMT0 internal drive.

The `--disable` command switches from recovery mode to normal boot mode. It is automatically invoked once the enable is finished and under most circumstances should not be manually invoked.

If run without either `--enable` or `--disable`, the command will print its help information.

Synopsis

```
$ cscli restore_mgmt [-h] [-s] [--enable | --disable]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-s |--show`	Displays the MGMT recovery status.
`|--enable`	Enables the MGMT recovery, boot MGMT normally.
`|--disable`	Disables the MGMT recovery, boot MGMT normally.

`show_new_nodes` Command

Modes: Daily

The `show_new_nodes` command displays a table of new OSS nodes and their resources.

Synopsis

```
$ cscli show_new_nodes [-h] [-v] |-c cluster_name
|--cluster cluster_name
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-v, --verbose`	Specifies the verbose mode.
`-c` *`cluster_name`*`|--cluster` *`cluster_name`*	*This parameter is deprecated.* It is supported only for backward compatibility.

17.10 `dm` and `dwd` Unresponsive Drives (UD) Commands

Modes: Daily

The `dm` and `dwd` commands have been deprecated in 2.1.0.

To address unresponsive drive issues, the diskmonitor tool was updated to be able to modify maximum reading errors threshold for arrays. Additionally, the DWD (disk watcher) daemon was created to allow to power down drives in case if they are failing or modify threshold for SCSI task-abort. The following commands let customers modify those values:

- `cscli dwd`
- `cscli dm`

Daemon Configuration Command

Modes: Daily

This command has been deprecated in 2.1.0.

The `cscli dwd` command may be used to configure disk watcher daemon (DWD) to allow or disallow it from powering down failed drives or to update SCSI task-abort threshold, where 0 sets to "ignore" those completely, '1' sets it to autocalculation mode and any other value will be treated as an actual numeric treshold.

Synopsis

```
$ cscli dwd [-h] [-n nodes] [--reset] [-l {yes,no}] [-a dwd_abrt_limit]
```

where:

Option	Description
`-h |--help`	Displays the help message and exits.
`-n nodes| --node` *`nodes`*	pdsh style node hostnames. Global configuration will be set without this argument.
`--reset`	Resets configuration to default values.
`-l {yes,no} | --lethal {yes,no}`	Allows (if set to 'yes') or disallows (if set to 'no') DWD to power down ("kill") failing drive. For GridRAID, this is the reconstruction rate.
`-a` *`dwd_abrt_limit`* `| --abort-limit` *`dwd_abrt_limit`*	Configures how DWD will treat SCSI task Aborts. This parameter expects integer (number) as an argument. Value of '0' will disable monitoring

Option	Description
	completely, value of '1' will enable auto-calculation and any other value will be treated as a threshold.

When called without arguments, current settings will be printed.

Default settings for DWD:

```
[root@snx11000n000 ~]# cscli dwd
------------------------------------------------------------
Node    Drive power-on on failure       Task abortion rate
------------------------------------------------------------
global              0                           1
Sample message for setting abort-task limit:
[root@snx11000n000 ~]# cscli dwd -a 15
dwd: done
Sample message for setting drive power down trigger:
[root@snx11000n000 ~]# cscli dwd -l yes
dwd: done
Sample output for DWD after setting new values:
[root@snx11000n000 ~]# cscli dwd
------------------------------------------------------------
Node    Drive power-on on failure       Task abortion rate
------------------------------------------------------------
global              1                           15
Those values can also be reset at any time similar to cscli dm command:
[root@snx11000n000 ~]# cscli dwd --reset
dwd: done
```

DM Service Configuration Management

The `cscli dm` command allows to update maximum read error threshold for all arrays or only specifically for RAID6 arrays.

This command has been deprecated in 2.1.0.

Synopsis

```
$ cscli dm [-h] [-n nodes] [--reset] [-g max_read_errs] [-m max_read_errs_r6]
```

where:

- `–g` is for global limits
- `–m` for setting RAID6 limits only

Additionally there is a “-reset” option to unset all thresholds to default. If no options were provided output would be similar to following:

Option	Description
`-h |--help`	Displays the help message and exits.
`-n nodes |--node` *`nodes`*	pdsh style nodes hostnames. Global configuration will be set without this argument.
`--reset`	Resets configuration to default values.

Option	Description
`-g` *`max_read_errs`* `| --global` *`max_read_errs`*	Sets maximum read errors value threshold before the drive is considered as failed. This is global limit for both GridRaid and MDRaid.
`-m` *`max_read_errs_r6`* `| --mdraid-limit` *`max_read_errs_r6`*	Sets max read errors value threshold before drive would be considered as failed. This is a MDRaid specific parameter to override the global limit if needed.

If no options were provided, output would be similar to:

```
[root@snx11000n00 ~]# cscli dm
---------------------------------------------------------------------
Node    Max read errors(Gridraid or mdraid)     Max read errors(mdraid(raid6)
specific)
---------------------------------------------------------------------
global              3500                        3500
```

Example output for changing values

Setting new limit for RAID6 arrays:

```
[root@snx11000n000 ~]# cscli dm -m 5000
dm: done
Setting new limit for all type of arrays:
[root@snx11000n000 ~]# cscli dm -g 7000
dm: done
Output after setting new values:
[root@snx11000n000 ~]# cscli dm
---------------------------------------------------------------------
Node    Max read errors(GridRAID or mdRAID)     Max read errors(mdraid(raid6)
specific)
---------------------------------------------------------------------
global              7000                        5000
```

To go back to default settings reset command may be used:

```
[root@snx11000n000 ~]# cscli dm --reset
dm: done
```

17.11 Filter Commands

The filter commands create, show, and delete a customer-nodes filter.

Create a Filter

Modes: Daily

The `create_filter` command creates a customer nodes filter.

Synopsis

```
$ cscli create_filter [-h] -i filter_sid -F filter_name -e filter_expr
```

Optional Arguments	Description
-h |--help	Displays the help message and exits.
-i *filter_sid* |--id *filter_sid*	Displays the symbol identifier of the filter.
-F *filter_name* |--name *filter_name*	Displays the filter name.
-e *filter_expr* |--expression *filter_expr*	Displays the filter expression. Examples: "host1,host2", "host[1-3]", "mds=primary".

Show Filters

Modes: Guest, Daily

The `show_filters` command shows all filters.

Synopsis

```
$ cscli show_filters [-h] [-P] [-C]
```

Optional Arguments	Description
-h |--help	Displays the help message and exits.
-P |--predefined	Displays only predefined filters.
-C |--custom	Displays only custom filters.

Delete a Filter

Modes: Daily

The `delete_filter` command deletes a customer nodes filter.

Synopsis

```
$ cscli delete_filter [-h] -i filter_sid
```

Optional Arguments	Description
-h |--help	Displays the help message and exits.
-i *filter_sid* |--id *filter_sid*	Displays the symbol identifier of the filter.

17.12 ibstat_check Command

Modes: Daily

The ibstat_check command, part of HA stack settings, disables and enables HA's probing of high speed network(s). This command is available for the "admin" account only.

Synopsis

```
$ cscli ibstat_check [-h] [-n nodes] (--enable | --disable)
```

Option	Description
-h |--help	Displays the help message and exits.
-n *nodes* |--node *nodes*	Displays pdsh-style nodes hostnames. **NOTE:** Lustre Server nodes only.
--enable	Enables HA's probing of the high speed network.
--disable	Disables HA's probing of the high-speed network.

17.13 lustre_users Commands

Modes: Daily

The advanced users (Lustre Users) commands support local user and group authentication commands. The following are available for Lustre users.

The lustre_users commands are used to configure the Lustre filesystem users settings. You can select the upcall method, configure services, and order user lookup across different services.

Synopsis

```
$ cscli lustre_users [-h] {ad,show,ldap,upcall,nis,apply,local,order}
```

If you are unable to connect, the following error message displays: “Unable to bind (connect) to (for example) an LDAP service using the parameters provided. Please check the URI (including port), Bind DN, and password.”

Positional Arguments	Description
show	Displays detailed information about all Lustre Users settings.
apply	Apply Lustre users configuration.
local	Define and manage local users and groups.
ldap	Management of LDAP settings.
ad	Management of AD settings.
nis	Management of NIS settings.
order	Instruct Lustre upcall to process user/group services in the order specified.
upcall	Configure the underlying method of Lustre upcall.
-h |--help	Displays the help message and exits.

17.13.1 `lustre_users ad` Service Commands

The `lustre_users ad` command is a subcommand of the `lustre_users` command. This command configures users and groups via an external AD service.

Synopsis

```
$ cscli lustre_users ad [-h] {clear,set,show}
```

Positional Arguments	Description
`show`	Displays AD settings.
`set`	Sets AD configuration.
`clear`	Removes all AD configurations.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

`lustre_users ad show` Subcommand

The `lustre_users ad show` command is a second-level subcommand of the `lustre_users ad` subcommand.

Synopsis

```
$ cscli lustre_users ad show [-h]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

`lustre_users ad set` Subcommand

The `lustre_users ad set` command is a subcommand of the `lustre_users ad` command.

Synopsis

```
$ cscli lustre_users ad set [-h] [-l ldap_uri] [-b base_dn] [-i bind_dn] [-p
password]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-l` *`ldap_uri`* `|--ldap-uri` *`ldap_uri`*	URI, for example: ldap://127.0.0.1:389.
`-b` *`base_dn`* `|--base-dn` *`base_dn`*	Base Domain Name.
`-i` *`bind_dn`* `|--bind-dn` *`bind_dn`*	Bind Domain Name.
`-p` *`password,`* `|--password` *`password`*	Bind password.

lustre_users ad clear Subcommand

The lustre_users ad clear command is a second-level subcommand of the lustre_users ad subcommand.

This command removes all AD configuration.

Synopsis

```
$ cscli lustre_users ad clear [-h] [-y]
```

Optional Arguments	Description
-h |--help	Displays the help message and exits.
-y |--yes	Confirms the configuration is cleared.

17.13.2 lustre_users apply Command

The lustre_users apply subcommand applies the Lustre users configuration.

Synopsis

```
$ cscli lustre_users apply [-h] [-y]
```

Optional Arguments	Description
-h |--help	Displays the help message and exits.
-y |--yes	Confirms the configuration is applied.

17.13.3 lustre_users ldap Subcommand

The lustre_users ldap command is a subcommand of the lustre_users command. This command configures users and groups via an external LDAP service. If you are unable to connect, the following error message displays: “Unable to bind (connect) to (for example) an LDAP service using the parameters provided. Please check the URI (including port), Bind DN, and password.”

Synopsis

```
$ cscli lustre_users ldap [-h] {clear,set,show}
```

where:

Positional Arguments	Description
show	Displays LDAP settings.
set	Sets LDAP configuration.
clear	Removes all LDAP configurations.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

lustre_users ldap set Subcommand

The `lustre_users ldap set` command is a second-level subcommand of the `lustre_users ldap` command.

IMPORTANT:

To properly configure LDAP using the `lustre_users ldap set` command, you must first add "directory" to the nss order:

```
$ cscli lustre_users order set local directory
```

To confirm the order, run the `lustre_users order show` command. See the `lustre_users order set` and `lustre_users_order_show` commands for more information.

Synopsis

```
$ cscli lustre_users ldap set [-h] [-l ldap_uri]
              [-b base_dn]  [-u user_dn] [-G group_dn]
              [-s hosts_dn] [-i bind_dn] [-p password]
              [--pvt-key filepath] [--tls-cert filepath] [--tls-ca filepath]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-l` *`ldap_uri`* `|--ldap-uri` *`ldap_uri`*	URI. For example: `ldap://127.0.0.1:389`
`-b` *`base_dn`* `|--base-dn` *`base_dn`*	Base domain name
`-u` *`user_dn`* `|--user-dn` *`user_dn`*	User domain name
`-G` *`group_dn`* `|--group-dn` *`group_dn`*	Group domain name
`-s` *`hosts_dn`* `|--hosts-dn` *`hosts_dn`*	Hosts domain name
`-i` *`bind_dn`* `|--bind-dn` *`bind_dn`*	Bind domain name
`-p` *`password,`* `|--password` *`password`*	Bind password.
`--pvt-key` *`filepath`*	Private key file (pem).
`--tls-cert` *`filepath`*	TLS certificate file (crt).
`--tls-ca` *`filepath`*	TLS ca certificate file (crt).

lustre_users ldap clear Subcommand

The `lustre_users ldap clear` command is a second-level subcommand of the `lustre_users ldap` subcommand. This command removes all LDAP configuration.

Synopsis

```
$ cscli lustre_users ldap clear [-h] [-y]
```

Optional Arguments	Description
-h |--help	Displays the help message and exits.
-y |--yes	Confirms the configuration is cleared.

lustre_users ldap show Subcommand

The lustre_users ldap show command is a second-level subcommand of the lustre_users ldap subcommand.

Synopsis

```
$ cscli lustre_users ldap show [-h]
```

where:

Optional Arguments	Description
-h |--help	Displays the help message and exits.

17.13.4 lustre_users local Commands

The lustre_users local command is a subcommand of the lustre_userscommand. This command defines users and groups locally, using files in the SysV standard formats of /etc/passwd and /etc/group.

Synopsis

```
$ cscli lustre_users local [-h] {clear,show,set_all,get_all}
```

Positional Arguments	Description
get_all	Downloads Lustre users/groups files.
show	Displays Lustre users/groups settings.
set_all	Uploads Lustre users/groups files.
clear	Resets Lustre users/groups to pristine state.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

lustre_users local get_all Subcommand

The lustre_users local get_all command is a second-level subcommand of the lustre_users local subcommand.

This command downloads the current locally-defined users and/or groups, as files in the SysVstandard, colon-separated form of /etc/passwd and /etc/group. If there are no locally-defined users or groups, a template file is downloaded.

Synopsis

```
$ cscli lustre_users local get_all [-h] [-u [users_file]] [-g [groups_file]] [-y]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-u [users_file]|--users-file [users_file]`	The name that is used to save the users file (default `lustre-users.txt`)
`-g [groups_file] |--groups-file [groups_file]`	The name that is used to save the groups file (default lustre=`groups.txt`)
`-y |--yes`	Overwrites an existing file of the same name without prompt.

`lustre_users local set_all` Subcommand

The `lustre_users local set_all` command is a second-level subcommand of the `lustre_users local` subcommand.

This command uploads a new user and/or group file, which will entirely replace the previous set of defined users and/or groups. It is strongly recommended to download the current definitions to a file, then add, delete, or edit entries in that file in order to upload changes.

> **IMPORTANT:** It is strongly recommended to download the current definitions to a file, then add, delete, or edit entries in that file in order to upload changes.

Synopsis

```
$ cscli lustre_users local set_all [-h] [-u users_file] [-g groups_file][-y]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-u users_file|--users-file users_file`	A path to a file in SysV '/etc/passwd' format.
`-g groups_file|--groups-file groups_file`	A path to a file in SysV '/etc/group' format.
`-y |--yes`	Overwrites an existing file of the same name without prompt.

`lustre_users local clear` Subcommand

The `lustre_users local clear` command is a second-level subcommand of the `lustre_users` local subcommand.

This command resets Lustre users/groups to pristine state.

Synopsis

```
$ cscli lustre_users local clear [-h] [-y]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-y |--yes`	Confirms the reset.

`lustre_users local show` Subcommand

The `lustre_users local show` command is a second-level subcommand of the `lustre_users` local subcommand.

This command resets displays the local users and groups.

Synopsis

```
$ cscli lustre_users local show [-h]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

Output:

```
# cscli lustre_users local show
Local:
  Number of Groups: 3
  Number of Users: 3
```

17.13.5 `lustre_users lookup` Command

The `lustre_users lookup` command is a subcommand of the `lustre_users` command. This command is used to look up a Lustre user or group. *(Introduced with 2.1.0.)*

Synopsis

```
$ cscli lustre_users lookup [-h] (-U user | -G group)
```

Optional Arguments	Description
`-Uuser |--user user`	username or uid
`-G group |--group group`	groupname or gid
`-h |--help`	Displays the help message and exits.

17.13.6 `lustre_users nis` Service Commands

The `lustre_users nis` command is a subcommand of the `lustre_users` command. This command configures users and groups via the Network Information Service (NIS).

Synopsis

```
$ cscli lustre_users nis [-h] {clear,set,show}
```

Positional Arguments	Description
`show`	Displays NIS settings.
`set`	Sets NIS configuration.

Positional Arguments	Description
`clear`	Removes NIS configuration.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

lustre_users nis set Subcommand

The `lustre_users nis set` command is a subcommand of the `lustre_users nis` command.

Synopsis

```
$ cscli lustre_users nis set [-h] [-s nis_server] [-d nis_domain]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-s` *`nis_server`* `|--nis_server` *`nis_server`*	Specifies this option 1 to 3 times using IP address or fully-qualified domain name.
`-d` *`nis_domain`* `|--nis_domain` *`nis_domain`*	NIS domain. Example: nisdomain.

lustre_users nis clear Subcommand

The `lustre_users nis clear` command is a second-level subcommand of the `lustre_users nis` subcommand.

This command removes all NIS configuration.

Synopsis

```
$ cscli lustre_users nis clear [-h] [-y]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-y |--yes`	Confirms the configuration is cleared.

lustre_users nis show Subcommand

The `lustre_users nis show` command is a second-level subcommand of the `lustre_users nis` subcommand.

Synopsis

```
$ cscli lustre_users nis show [-h]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

17.13.7 lustre_users order Commands

The lustre_users order subcommand configures Lustre filesystem users. The command selects upcall method, configure services, and order for user lookup across different services. This command instructs Lustre upcall to process user or group services in the order specified.

Synopsis

```
$ cscli lustre_users order [-h] {set,show}
```

Positional Arguments	Description
show	Displays the order for looking up services when more than one is defined.
set	Instructs Lustre upcall to process user/group services in the order specified.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

lustre_users order set Subcommand

The lustre_users order set command is a second-level subcommand of the lustre_users command. This command sets the order for looking up services when more than one is defined.

Synopsis

```
$ cscli lustre_users order set [-h] {local,nis,directory}
                                    [{local,nis,directory} ...]
```

Positional Arguments	Description
local,nis,directory	Sets local, NIS and directory services.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

lustre_users order show Subcommand

The lustre_users order show command is a second-level subcommand of the lustre_users command.

This command shows the order for looking up services when more than one is defined.

Synopsis

```
$ cscli lustre_users order show [-h]
```

Optional Arguments	Description
-h |--help	Displays the help message and exits.

17.13.8 `lustre_users show` Subcommand

The `lustre_users show` command is a subcommand of the `lustre_users` command. This command shows detailed information about all Lustre Users settings.

Synopsis

```
$ cscli lustre_users show [-h]
```

where:

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

Example:

```
# cscli lustre_users show
Lustre Users
   Upcall: generic
   Order: local, directory
   LDAP:
      Servers: ldap://dc.xyua:3268
      Base DN: dc=dc,dc=xyua
      Bind DN: administrator@dc.xyua
      Password: ********
      User DNs: cn=Users,dc=dc,dc=xyua
      Group DNs: not defined
      Hosts DNs: not defined
      TLS Cert: TLS cert is valid.
         Version: 3
         Subject: CN=cfw-dc0.cfw-ad.no-more.kiev.ua
         Issuer: DC=ua, DC=kiev, DC=no-more, DC=cfw-ad, CN=ca
         Serial: 106310341937692867035147
      TLS Private Key: TLS pvtkey is valid, 1024 bits
      TLS CA Cert: TLS CA cert is valid.
         Version: 3
         Subject: C=UA, ST=Ukraine, L=Kiev, O=No More BBS, OU=Software Department,
CN=NoMore Root CA v3/emailAddress=ca@no-more.kiev.ua
         Issuer: C=UA, ST=Ukraine, L=Kiev, O=No More BBS, OU=Software Department,
CN=NoMore Root CA v3/emailAddress=ca@no-more.kiev.ua
         Serial: 9313703399412187718
   NIS: (not configured)
   Local: (not configured)
```

17.13.9 `lustre_users upcall` Authentication Subcommand

The `lustre_users upcall` subcommand configures the underlying method of Lustre upcall. The `lustre_users upcall` command must have a unique user ID. The administrator receives a warning if the number of UIDs exceeds 5000 users. This operation does not fail if the threshold is exceeded.

Synopsis

```
$ cscli lustre_users upcall [-h] {set,show}
```

Positional Arguments	**Description**
`-show`	Displays the underlying method of Lustre upcall.
`-set`	Configures the underlying method of Lustre upcall.

Optional Arguments	**Description**
`-h |--help`	Displays the help message and exits.

`lustre_users upcall set` Subcommand

The `lustre_users upcall set` command is a second-level subcommand of the `lustre_users` command. Select the underlying upcall method for Lustre users and groups: '`ad_only`' for Active Directory only, or '`generic`' for any of LDAP, NIS, or locally defined users and groups.

The Lustre file system must be unmounted in order to change upcall type.

Synopsis

```
$ cscli lustre_users upcall set [-h] -u {ad_only,generic,none}
```

Optional Arguments	**Description**
`-h |--help`	Displays the help message and exits.
`-u {ad_only,generic}| --upcall {ad_only,generic,none}`	The `ad_only` command enables only Active Directory users and groups only. The `generic` command allows for any of LDAP, NIS, or locally-defined users and groups. The `none` command disables all Lustre upcall commands, which may result in file permission errors.

`lustre_users upcall show` Subcommand

The `lustre_users upcall show` command is a second-level subcommand of the `lustre_users` subcommand.

This command shows the underlying method of Lustre upcall.

Synopsis

```
$ cscli lustre_users upcall show [-h]
```

Optional Arguments	**Description**
`-h |--help`	Displays the help message and exits.

17.13.1 New and Old Lustre Users Commands Comparison 0

Current Commands (Starting with Release 1.5.0)	Old Commands (Prior to Release 1.5.0)
`lustre_users`	None

`lustre_users show`	`get_lustre_users_ldap`
	`get_lustre_users_nis`
	`get_lustre_users_upcall`

`lustre_users local`	None
`lustre_users ldap set`	`set_lustre_users_ldap`
`lustre_users ldap clear`	`set_lustre_users_ldap -C`
`lustre_users nis set`	`set_lustre_users_nis`
`lustre_users nis clear`	`set_lustre_users_nis -c`
`lustre_users upcall`	`set_lustre_users_upcall`
`lustre_users order`	None
`lustre_users apply`	None
`lustre_users ad set`	`set_lustre_users_ad`

17.14 `Lustre` Management Command

The `lustre` command is used for Lustre management functions, such as enabling, disabling, and changing Lustre changelog flags.

Synopsis

```
$ cscli lustre [-h] {changelog}
```

Where:

Positional Arguments	Description
`changelog`	Manage Lustre changelog configuration.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

`lustre changelog` Subcommand

The `changelog` command is a subcommand of the `lustre` command. Use the command to enable and disable Lustre changelog, and to change and reset the changelog flags.

Synopsis

```
$ cscli lustre changelog [-h] {reset,enable,change,disable,show}
```

Where:

Positional Arguments	Description
show	Show Lustre changelog status.
enable	Enable Lustre changelog.
disable	Disable Lustre changelog.
change	Change Lustre changelog flags.
reset	Reset Lustre changelog flags to default.

Optional Arguments	Description
-h |--help	Displays the help message and exits

lustre changelog show Subcommand

The lustre changelog show command is a second-level subcommand of the lustre command.

Synopsis

```
$ cscli lustre changelog show [-h]
```

Where:

Optional Arguments	Description
-h |--help	Displays the help message and exits

lustre changelog enable Subcommand

The lustre changelog enable command is a second-level subcommand of the lustre command.

Synopsis

```
$ cscli lustre changelog enable [-h] [-f [FlAG [FlAG ...]]]
                                     [-t [TARGET [TARGET ...]]]
```

Where:

Optional Arguments	Description
-f [FLAG[FLAG ...]] |--flag [FLAG [FLAG ...]]	Lustre changelog flags.
-t [TARGET[TARGET ...]]	Lustre targets.

Optional Arguments	Description
`|--target [TARGET [TARGET ...]]`	
`-h |--help`	Displays the help message and exits.

lustre changelog disable Subcommand

The `lustre changelog disable` command is a second-level subcommand of the `lustre` command.

Synopsis

```
$ cscli lustre changelog disable [-h] [-t [target [target ...]]]
```

Where:

Optional Arguments	Description

`-t [target [target ...]]`
`|--target [target [target ...]]`
	Lustre targets.

`-h |--help`	Displays the help message and exits

lustre changelog change Subcommand

The `lustre changelog change` command is a second-level subcommand of the `lustre` command.

Synopsis

```
$ cscli lustre changelog change [-h] [-f [flag [flag ...]]]
                                [-t [target [target ...]]]
```

Where:

Optional Arguments	Description

`-f [flag [flag ...]]`
`|--f [flag [flag ...]]`
	Lustre changelog flags.

`-t [target [target ...]]`
`|--target [target [target ...]]`
	Lustre targets.

`-h |--help`	Displays the help message and exits.

lustre changelog reset Subcommand

The `lustre changelog reset` command is a second-level subcommand of the `lustre` command.

Synopsis

```
$ cscli lustre changelog reset [-h] [-t [target [target ...]]]
```

Where:

Optional Arguments	Description

`-t [target[target ...]]`
`|--target [target [target ...]]`
	Lustre targets.

`-h |--help`	Displays the help message and exits.

17.15 lustre_network Commands

Modes: Daily

The `lustre_network` commands are used for Hidden Service Protocol (HSP) IP address management.

Synopsis

```
$ cscli lustre_network [-h]
{list_ranges,find_gaps,list_hosts,add_range,extend_range,remove_range,defaults}
```

where:

Positional Arguments	Description
`list_ranges`	List ranges of IPs for InfiniBand fabric
`find_gaps`	Show unused IPs in ranges of InfiniBand fabric
`list_hosts`	Shows the network pairing information (ip and lhostname) of data_network of the cluster.
`add_range`	Add new range of IPs into InfiniBand fabric
`extend_range`	Extend range of IPs for InfiniBand fabric
`remove_range`	Remove range of IPs for InfiniBand fabric
`defaults`	Manage default gateway and netmask for Data Network.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits

list_ranges Subcommand

The `list_ranges` command is a subcommand of the `lustre_network` command.

Synopsis

```
$ cscli lustre_network list_ranges [-h]
```

where:

Optional Arguments	Description
-h |--help	Displays the help message and exits

find_gaps Subcommand

The find_gaps command is a subcommand of the lustre_network command.

Synopsis

```
$ cscli lustre_network find_gaps [-h]
```

where:

Optional Arguments	Description
-h |--help	Displays the help message and exits

add_range Subcommand

The add_range command is a subcommand of the lustre_network command.

Synopsis

```
$ cscli lustre_network add_range [-h] -f from_ip -t to_ip [-h]
```

where:

Optional Arguments	Description
-h |--help	Displays the help message and exits
-f *from_ip* | --from_ip *from_ip*	Displays first IP address in the range
-t *to_ip* | --to_ip *to_ip*	Displays last IP address in the range

extend_range Subcommand

The extend_range command is a subcommand of the lustre_network command. Use this command to extend a range of IP addresses for the HSN.

Synopsis

```
$ cscli lustre_network extend_range [-h] -i id [-f from_ip] [-t to_ip] [-h]
```

where:

Optional Arguments	Description
-h |--help	Displays the help message and exits
-i *id* | --id *id*	Displays the range ID

Optional Arguments	Description
`-f` *from_ip* `|` `--from_ip` *from_ip*	Displays first IP address in the range
`-t` *to_ip* `|` `--to_ip` *to_ip*	Displays last IP address in the range

remove_range Subcommand

The `remove_range` command is a subcommand of the `lustre_network` command. Use this command to remove a range of IP addresses for the HSM.

Synopsis

```
$ cscli lustre_network remove_range [-h]
```

where:

Optional Arguments	Description
`-h | --help`	Displays the help message and exits
`-i id | --id` *id*	Displays the range ID

defaults Subcommand

The `defaults` command is a subcommand of the `lustre_network` command. Use the command to manage the default gateway and netmask for the Data Network.

Synopsis

```
$ cscli lustre_network defaults [-h]
```

where:

Optional Arguments	Description
`-h |--help`	Displays the help message and exits

17.16 lustre_perf Command

Modes: Daily

The `lustre_perf` (Lustre Performance) command is used to view the Lustre file system performance in the Sonexion system (*Deprecated in 3.0.0*).

Synopsis

```
$ cscli lustre_perf  [-h] {fetch,ltop,list,status,abort,clean}
```

where:

Positional Arguments	Description
`fetch`	This command exports historical Lustre data between start time and end time to the local filesystem. Use `cscli lustre_perf list` to find the location of the resulting output.
`ltop`	Displays live information about a Lustre file system. Use the `--help` command for more details.
`list`	Lists the full path of any existing log files. Use the `--help` command for more details.
`status`	Returns the status of the last run command (or the currently running command if it is non-blocking and a process is still running). Use the `--help` command for more details.
`abort`	Aborts the currently running export job. Use the `--help` command for more details.
`clean`	Deletes all export files in the export folder. Use the `--help` command for more details

Optional Arguments	Description
`|--version`	Displays twisted version and exits.
`-h |--help`	Displays the help message and exits

`lustre_perf fetch` Subcommand

The `lustre_perf fetch` command is a subcommand of the `lustre_perf` command, and is used to export historical Lustre data.

Synopsis

```
$ cscli lustre_perf fetch [-h] [-s start] [-e end]
```

where:

Optional Arguments	Description

`-s` *`start`*
`|--starttime` *`start`*
	Specify the DATETIME (YYYY-MM-DDThh:mm:ss(+|-)hh:mm) to start fetching data from.

`-e` *`end`*
`|--endtime` *`end`*
	Specify the ENDTIME (YYYY-MM-DDThh:mm:ss(+|-)hh:mm) to stop fetching data.

`|--version`	Displays twisted version and exits.
`-h |--help`	Displays the help message and exits.

Examples

Normal Usage:

```
$ cscli lustre_perf fetch -s 2015-03-01T08:25:20-00:00 -e 2015-03-
02T08:29:00-00:00
```

Lustre performance statistics export started.

Use `lustre_perf status` to monitor progress and `lustre_perf list` to find the resulting output.

> **IMPORTANT:** During fetching, the user can run `cscli lustre_perf status` in a separate console to see fetch status. See the `status` subcommand for more details.

Error Usage: Fetch Interval Is Too Small

```
$ cscli lustre_perf fetch -s 2015-03-01T08:25:20-00:00 -e 2015-03-
01T08:26:00-00:00
Error! Interval between start time and end time need to be 120 seconds minimum.
```

Error Usage: Fetch While Other Fetch/Clean Process Is Running

```
$ cscli lustre_perf fetch -s 2015-03-01T08:25:20-00:00 -e 2015-03-
02T08:29:00-00:00
Error! Unable to acquire lock /tmp/lustre_perf.pid
Reason: Only one instance of 'lustre_perf fetch/clean' is allowed at once.
```

Error Usage: Fetch Without Enough Parameters

```
$ cscli lustre_perf fetch
```

`lustre_perf ltop` Subcommand

The `lustre_perf ltop` command is a subcommand of the `lustre_perf` command, and is used to display live information about a Lustre file system.

Synopsis

```
$ cscli lustre_perf ltop [-h] [-n] [-f=filter]
```

where:

Optional Arguments	Description
`-n` `|--no-summary`	Omit summary of file system stats in output.

`-f=`*`filter`*
`|--filter=`*`filter`*
	Filter by regular expression of target name. [default:]

`|--version`	Displays twisted version and exits
`-h |--help`	Displays the help message and exits

Examples

Normal Usage

```
$ cscli lustre_perf ltop
Filesystem: testfs
  Inodes:     4259.059m total,    1625.000 used (0%),    4259.058m free
  Space:       134.845t total,    1387.531g used (1%),    133.458t free
     Bytes/s:    8234.887m read,    0.000 write,         8234.887m IOPS
    MDops/s:       0 open,          0 close,      0 getattr,    0 setattr
                     0 link,          0 unlink,      0 mkdir,      0 rmdir
                     0 statfs,           0 rename,     0 getxattr
>MDT
0000    MDS:        nsit203,        0 %cpu,                    2 %mem
  Inodes:    2016.608m total,    217.000 used (0%),      2016.608m free
  Space:    3096.947g total,    4396.978m used  (0%),    3092.550g free
  MDops/s:    0 open,            0 close,     0 getattr,    0 setattr
              0 link,            0 unlink,    0 mkdir,      0 rmdir
              0 statfs,          0 rename,    0 getxattr
0001    MDS:        nsit206,        2 %cpu,            3 %mem
  Inodes:      1062.014m total,     205.000 used (0%),    1062.013m free
  Space:    3805.960g total,    4369.818m used (0%),    3801.591g free
  MDops/s:    0 open,            0 close,        0 getattr,    0 setattr
              0 link,            0 unlink,       0 mkdir,      0 rmdir
              0 statfs,          0 rename,       0 getxattr
0002    MDS:        nsit207,        2 %cpu,            3 %mem
  Inodes:      1062.014m total,    205.000 used (0%),    1062.013m free
  Space:    3805.960g total,    4369.822m used (0%),    3801.591g free
  MDops/s:    0 open,              0 close,        0 getattr,    0 setattr
              0 link,              0 unlink,       0 mkdir,      0 rmdir
              0 statfs,            0 rename,       0 getxattr

>OST    S    OSS        rB/s    wB/s   %cpu     %mem    %spc
0000    -    nsit204    4102m    0    27    20    1
0001    -    nsit205    4133m    0    25    15    1
```

Normal Usage – No Summary

```
$ cscli lustre_perf ltop --no-summary
>MDT
0000    MDS:    nsit203,        0 %cpu,                         2 %mem
  Inodes:    2016.608m total,    217.000 used (0%),    2016.608m  free
  Space:     3096.947g total,    4396.978m used (0%),    3092.550g free
  MDops/s:      0 open,     0 close,      0 getattr,      0 setattr
                0 link,     0 unlink,     0 mkdir,        0 rmdir
                0 statfs,      0 rename,     0 getxattr
0001    MDS:        nsit206,        2 %cpu,            3 %mem
  Inodes:    1062.014m total,    205.000 used (0%),      1062.013m free
  Space:     3805.960g total,    4369.818m used (0%),    3801.591g free
  MDops/s:      0 open,      0 close,      0 getattr,      0 setattr
                0 link,      0 unlink,     0 mkdir,        0 rmdir
                0 statfs,    0 rename,     0 getxattr
0002    MDS:        nsit207,        2 %cpu,            3 %mem
  Inodes:    1062.014m total,    205.000 used (0%),           1062.013m free
  Space:     3805.960g total,    4369.822m used (0%),    3801.591g free
  MDops/s:    0 open,      0 close,         0 getattr,    0 setattr
              0 link,      0 unlink,          0 mkdir,      0 rmdir
              0 statfs,    0 rename,          0 getxattr

>OST    S    OSS     rB/s    wB/s   %cpu    %mem    %spc
0000    -    nsit204   4102m    0    27    20     1
0001    -    nsit205   4133m    0    25    15     1
```

Normal Usage - Filter by Device Name

```
$ cscli lustre_perf ltop --filter=OST0000
Filesystem:  testfs
  Inodes:    4259.059m total,    1625.000 used (0%),    4259.058m free
  Space:      134.845t total,    1387.531g used (1%),    133.458t free
  Bytes/s:    8234.887m read,    0.000 write,         8234.887m IOPS
  MDops/s:    0 open,      0 close,     0 getattr,    0 setattr
           0 link,      0 unlink,    0 mkdir,       0 rmdir
           0 statfs,    0 rename,    0 getxattr
>OST S    OSS     rB/s    wB/s    %cpu    %mem    %spc
0000 -    nsit204   4102m    0        27      20      1

$ cscli lustre_perf ltop --filter=MDT
Filesystem:    testfs
  Inodes:    4259.059m total,    1625.000 used (0%),    4259.058m free
  Space:      134.845t total,    1387.531g used (1%),    133.458t free
  Bytes/s:    8234.887m read,     0.000 write,        8234.887m IOPS
  MDops/s:    0 open,      0 close,     0 getattr,    0 setattr
              0 link,      0 unlink,    0 mkdir,      0 rmdir
              0 statfs,    0 rename,    0 getxattr
>MDT
0000         MDS:     nsit203,          0 %cpu,         2 %mem
  Inodes:    2016.608m total,    217.000 used (0%),     2016.608m  free
  Space:     3096.947g total,      4396.978m used (0%),    3092.550g  free
  MDops/s:    0 open,      0 close,     0 getattr,       0 setattr
              0 link,      0 unlink,    0 mkdir,         0 rmdir
              0 statfs,    0 rename,    0 getxattr
0001    MDS:        nsit206,        2 %cpu,        3 %mem
  Inodes:    1062.014m total,    205.000 used (0%),      1062.013m free
  Space:    3805.960g total,    4369.818m used (0%),       3801.591g free
  MDops/s:    0 open,      0 close,      0 getattr,       0 setattr
              0 link,      0 unlink,     0 mkdir,         0 rmdir
              0 statfs,    0 rename,     0 getxattr
0002    MDS:        nsit207,        2 %cpu,        3 %mem
  Inodes:    1062.014m total,    205.000 used (0%),       1062.013m free
  Space:    3805.960g total,    4369.822m used (0%),       3801.591g free
  MDops/s:    0 open,      0 close,      0 getattr,       0 setattr
              0 link,      0 unlink,     0 mkdir,         0 rmdir
              0 statfs,    0 rename,     0 getxattr
```

lustre_perf list Subcommand

The lustre_perf list command is a subcommand of the lustre_perf command, and is used to list the full path of any existing log files.

Synopsis

```
$ cscli lustre_perf list [-h]
```

where:

<table>
<tr><th>Optional Arguments</th><th>Description</th></tr>
<tr><td>|--version</td><td>Displays twisted version and exits</td></tr>
<tr><td>-h |--help</td><td>Displays the help message and exits</td></tr>
</table>

Examples

Normal Usage - When No Fetch Results Exist:

```
$ cscli lustre_perf list
No files found.
```

Normal Usage - When Fetch Results Exist:

```
$ cscli lustre_perf list
/mnt/mgmt/var/lib/lustre_perf/data/20150301082520000_20150301092600000.csv.gz
/mnt/mgmt/var/lib/lustre_perf/data/20150225100025000_20150302183113000.csv.gz
Total: 21.81 MB used by 2 files
```

lustre_perf status Subcommand

The `lustre_perf status` command is a subcommand of the `lustre_perf` command, and is used to return the status of the last command run (or the currently running command if it is non-blocking and a process is still running).

Synopsis

```
$ cscli lustre_perf status [-h]
```

where:

Optional Arguments	Description
|--version	Displays twisted version and exits
-h |--help	Displays the help message and exits

Examples

When No Fetching Record is Detected:

```
$ cscli lustre_perf status
Status: no files
Message: 'cscli lustre_perf fetch' has never been run, or 'cscli
lustre_perf clean' was recently called to purge all export files
```

When Fetching Is In Progress:

```
$ cscli lustre_perf status
Status: running - fetch for data from 2015-03-01T08:25:20+00:00 to
2015-03-01T09:26:00+00:00
```

When a Previous Fetching Command Succeeded:

```
$ cscli lustre_perf status
Status: completed - fetch for data from 2015-03-01T08:25:20+00:00 to
2015-03-01T09:26:00+00:00
Filename:
/mnt/mgmt/var/lib/lustre_perf/data/20150301082520000_20150301092600
0000.csv.gz
Message: Successfully completed fetching data from 2015-03-
01T08:25:20+00:00 to 2015-03-01T09:26:00+00:00
```

When a Previous Fetching Command Failed:

```
$ cscli lustre_perf status
Status: error
Message: Timeout Error: please try again later
```

lustre_perf clean Subcommand

The lustre_perf clean command is a subcommand of the lustre_perf command, and is used to delete all export files in the export folder.

Synopsis

```
$ cscli lustre_perf clean [-h]
```

where:

Optional Arguments	Description
|--version	Displays twisted version and exits
-h |--help	Displays the help message and exits

Examples

Normal Usage:

```
$ cscli lustre_perf clean
No archives to clean

$ cscli lustre_perf clean
2 archives with size 4.12 KB were successfully removed

$ sudo lustre_perf list
/mnt/mgmt/var/lib/lustre_perf/data/20150303204620000_201503032346200
000.csv.gz
/mnt/mgmt/var/lib/lustre_perf/data/20150304120000000_201503041500000
000.csv.gz
Total: 3.27 KB used by 2 files
$ sudo lustre_perf clean
$ sudo lustre_perf list
No files found.
$
```

ErrorUsage -- When Fetch/Clean Commands are Running:

```
$ cscli lustre_perf clean
Error: lustre_perf clean failed.
Reason: lustre_perf is currently running and creating output
```

17.17 monitor Command for System Health

Modes: Site configuration, Guest, Daily

The monitor command monitors and displays current health and status information for the cluster nodes and elements.

Synopsis

```
$ cscli monitor [-h] {nodes,elements,health} ...
```

Positional Arguments	Description
`health`	Current overall health information and status summary.
`nodes`	Current status for nodes.
`elements`	Current status for elements.

Optional Arguments	Description
`-h | --help`	Displays the help message and exits.

`monitor nodes` Subcommand

The `monitor nodes` command monitors individual nodes.

Synopsis

```
$ cscli monitor nodes [-h] [-y] [-v] [-n node_spec | -g genders_query] [-N
{down,unreachable,up,pending}]
```

Description	Positional Arguments
`-h | --help`	Displays the help message and exits.
`-y | --yaml`	Displays output data in YAML format.
`-v | --verbose`	Outputs extra data.
`-n` *`node_spec`* `| --node` *`node_spec`*`| --nodes` *`node_spec`*	Looks through passed hostname elements. Looks for pdsh style nodes host names (e.g. `node[100-110,120]`).
`-g` *`genders_query`*	Displays the node genders attributes query (e.g. mds=primary).
`-N {down,unreachable,up,pending} | --nodestatus {down,unreachable,up,pending}`	Displays node status.

`monitor elements` Subcommand

The `monitor elements` command monitors individual nodes.

Synopsis

```
$ cscli monitor elements [-h] [-y] [-v] [-n node_spec | -g genders_query] [-N
{down,unreachable,up,pending}] [-U {unknown,warning,ok,critical,pending}] [-S
element_filter]
```

Positional Arguments	Description
`-h | --help`	Displays the help message and exits.
`-y | --yaml`	Displays output data in YAML format.
`-v | --verbose`	Outputs extra data.

Positional Arguments	Description
`-n` *`node_spec`*`|` `--node` *`node_spec`*`|` `--nodes` *`node_spec`*	Looks through passed hostname elements. Looks for pdsh-style nodes host names (e.g. `node[100-110,120]`).
`-g` *`genders_query`*	Displays the node genders attributes query (e.g. mds=primary).
`-N {down,unreachable,up,pending} | --nodestatus {down,unreachable,up,pending} node status.`	Displays node status.
`-U {unknown,warning,ok,critical,pending} | --elementstatus {unknown,warning,ok,critical,pending}`	Displays element status.
`-S` *`element_filter`* `| --search` *`element_filter`*	Searches by element name. The pattern is case-sensitive. Regular expressions are allowed.

> **IMPORTANT:** Calling this command with no options may result in thousands of elements on a large system.

17.17.1 `monitor` Command Output Examples

Following are examples of outputs resulting from use of the `monitor` command, with outputs including OK, WARNING and CRITICAL.

```
[root@cls12345n000 ~]# cscli monitor health
Nodes:
up: 8   down: 0         unreachable: 0  pending: 0      total: 8
Elements:
ok: 78  warning: 0      critical: 0     unknown: 0      pending: 0      total: 78
```

No output means no errors :

```
[root@cls12345n000 ~]# cscli monitor elements -U unknown -U pending -U warning -U critical
[root@cls12345n000 ~]# cscli monitor elements -U unknown
[root@cls12345n000 ~]# cscli monitor elements -U pending
[root@cls12345n000 ~]# cscli monitor elements -U critical
[root@cls12345n000 ~]# cscli monitor elements -U warning
[root@cls12345n000 ~]# cscli monitor nodes
cls12345n000:  UP  for 28d 15h 33m 26s  checked 2014-02-06 10:00:36  "PING OK - Packet loss = 0%, RTA = 0.03 ms"
cls12345n001:  UP  for 15d 18h 14m 44s  checked 2014-02-06 10:02:56  "PING OK - Packet loss = 0%, RTA = 0.16 ms"
cls12345n002:  UP  for 15d 18h  6m 54s  checked 2014-02-06 10:05:36  "PING OK - Packet loss = 0%, RTA = 0.18 ms"
cls12345n003:  UP  for 15d 18h  8m 54s  checked 2014-02-06 10:03:36  "PING OK - Packet loss = 0%, RTA = 0.18 ms"
cls12345n003-Enclosure-R1C1-21U:  UP  for 28d 15h 35m  8s  checked 2014-02-06 09:55:36  "OK"
cls12345n004:  UP  for 5d 17h 14m 44s  checked 2014-02-06 10:01:26  "PING OK - Packet loss = 0%, RTA = 0.16 ms"
cls12345n005:  UP  for 5d 18h 30m 14s  checked 2014-02-06 10:03:26  "PING OK - Packet loss = 0%, RTA = 0.19 ms"
cls12345n005-Enclosure-R1C1-5U:  UP  for 28d 15h 34m 12s  checked 2014-02-06 10:02:36  "OK"
[root@cls12345n000 ~]# cscli monitor elements
```

Subset of output:

```
cls12345n000  "Arrays and Disk Status":  OK  for 28d 15h 33m 49s  checked 2014-02-06 10:00:52  "All arrays are operating normally"
cls12345n000  "Current Load":  OK  for 28d 15h 33m 49s  checked 2014-02-06 10:00:52  "OK - load average: 0.08, 0.03, 0.02"
cls12345n000  "Current Users":  OK  for 28d 15h 33m 49s  checked 2014-02-06 10:00:52  "USERS OK - 1 users currently logged in"
cls12345n000  "Free Space":  OK  for 21d 18h 45m 53s  checked 2014-02-06 10:00:52  "DISK OK - free space: / 181915 MB (98% inode=99%): /mnt/mgmt 778774 MB (99%
inode=99%):"
cls12345n000  "Network statistics":  OK  for 28d 15h 33m 49s  checked 2014-02-06 10:00:52  "NET OK - (Rx/Tx) eth0=(8.4B/5.6B), eth1=(535.5B/349.9B), eth2=(0.0B/0.0B),
eth3=(0.0B/0.0B), eth4=(0.0B/0.0B), ib0=(11.4B/0.0B), lo=(9.4B/9.4B), meth0=(8.4B/5.6B), meth1=(0.0B/0.0B)"
cls12345n000  "RAM usage":  OK  for 28d 15h 33m 49s  checked 2014-02-06 10:00:52  "OK - 11.6% (3807704 kB) used."
cls12345n000  "Swap Usage":  OK  for 28d 15h 33m 49s  checked 2014-02-06 10:00:52  "SWAP OK - 100% free (31999 MB out of 31999 MB)"
cls12345n000  "Total Processes":  OK  for 28d 15h 33m 49s  checked 2014-02-06 10:00:52  "PROCS OK: 407 processes with STATE = RSZDT"
cls12345n000  "crmd cpu usage":  OK  for 28d 15h 33m 49s  checked 2014-02-06 10:00:52  "OK - Process: crmd, User: 496, CPU: 0.0%, RAM: 0.0%, Start: Jan21, CPU Time:
127 min"
cls12345n000  "crmd memory usage":  OK  for 28d 15h 33m 49s  checked 2014-02-06 10:00:52  "OK - Process: crmd, User: 496, CPU: 0.0%, RAM: 0.0%, Start: Jan21, CPU
Time: 127 min"
cls12345n000  "heartbeat cpu usage":  OK  for 28d 15h 33m 49s  checked 2014-02-06 10:00:52  "OK - Process: heartbeat, User: root, CPU: 0.0%, RAM: 0.0%, Start: Jan21,
CPU Time: 695 min"
cls12345n000  "heartbeat memory usage":  OK  for 28d 15h 33m 49s  checked 2014-02-06 10:00:52  "OK - Process: heartbeat, User: root, CPU: 0.0%, RAM: 0.0%, Start:
Jan21, CPU Time: 695 min"
cls12345n000  "stonithd cpu usage":  OK  for 28d 15h 33m 49s  checked 2014-02-06 10:00:52  "OK - Process: stonithd, User: root, CPU: 0.0%, RAM: 0.0%, Start: Jan21,
CPU Time: 60 min"
cls12345n000  "stonithd memory usage":  OK  for 28d 15h 33m 49s  checked 2014-02-06 10:00:52  "OK - Process: stonithd, User: root, CPU: 0.0%, RAM: 0.0%, Start: Jan21,
CPU Time: 60 min"
cls12345n004  "Arrays and Disk Status":  OK  for 2d 14h 28m 10s  checked 2014-02-06 10:03:42  "All arrays are operating normally"
cls12345n004  "Current Load":  OK  for 28d 15h 32m 14s  checked 2014-02-06 10:03:42  "OK - load average: 0.01, 0.01, 0.01"
cls12345n004  "Current Users":  OK  for 28d 15h 32m 14s  checked 2014-02-06 10:03:42  "USERS OK - 0 users currently logged in"
cls12345n004  "Free Space":  OK  for 28d 15h 32m 14s  checked 2014-02-06 10:03:42  "DISK OK - free space: /tmp 15966 MB (99% inode=99%):"
cls12345n004  "Lustre Health":  OK  for 28d 15h 32m 14s  checked 2014-02-06 10:03:42  "OK:Lustre is ok"
cls12345n004  "Network statistics":  OK  for 28d 15h 32m 14s  checked 2014-02-06 10:03:42  "NET OK - (Rx/Tx) eth0=(16.9B/5.8B), ib0=(169.5B/60.9B), ib1=(0.0B/0.0B),
lo=(140.2B/140.2B), meth0=(16.9B/5.8B), meth1=(0.0B/0.0B), xyvnic0=(71.5B/75.2B)"
cls12345n004  "RAM usage":  OK  for 28d 15h 32m 14s  checked 2014-02-06 10:03:42  "OK - 12.9% (4203984 kB) used."
cls12345n004  "Swap Usage":  OK  for 28d 15h 32m 14s  checked 2014-02-06 10:03:42  "SWAP OK - 100% free (16386 MB out of 16386 MB)"
cls12345n004  "Total Processes":  OK  for 28d 15h 32m 14s  checked 2014-02-06 10:03:42  "PROCS OK: 1239 processes with STATE = RSZDT"
root@cls12345n000 ~]# cscli monitor elements -v
```

Subset of output:

```
cls12345n000  "Arrays and Disk Status":  OK  for 28d 15h 34m 45s  checked 2014-02-06 10:05:52  "All arrays are operating normally
Array: md64, status: Ok, t10: disabled
Total number of disk slots available: 24
Total number of disks found: 24
slot:   2, wwn: 5000c50043b1e71f, cap:         450098159616, dev:   sdl, parts:        0, status: Hot Spare, t10: 11110100000
slot:  21, wwn: 5000c500479061af, cap:         450098159616, dev:   sdv, parts:        0, status: Hot Spare, t10: 11110100000
MD RAID to Lustre mapping
Array /dev/md/cls12345n003:md64 doesn't have associated WIB array
Degraded Array information:
All arrays are in clean state on node cls12345n000"
Performance Data:  None
Current Attempt:  1/3 (HARD state)
Check Type:  passive
Check Latency / Duration:  None / 0.0
Next Scheduled Active Check:  None
Last State Change:  2014-01-08 18:32:24
Last Update:  2014-02-06 10:07:06
------------------------------------------------------------------------
cls12345n000  "Current Load":  OK  for 28d 15h 34m 45s  checked 2014-02-06 10:05:52  "OK - load average: 0.01, 0.02, 0.02"
Performance Data:  load1=0.013;1000000.000;1000000.000;0; load5=0.023;1000000.000;1000000.000;0; load15=0.020;1000000.000;1000000.000;0;
Current Attempt:  1/3 (HARD state)
Check Type:  passive
Check Latency / Duration:  None / 0.0
Next Scheduled Active Check:  None
Last State Change:  2014-01-08 18:32:24
Last Update:  2014-02-06 10:07:06
------------------------------------------------------------------------
cls12345n000  "Current Users":  OK  for 28d 15h 34m 45s  checked 2014-02-06 10:05:52  "USERS OK - 1 users currently logged in"
Performance Data:  users=1;10;50;0
Current Attempt:  1/3 (HARD state)
Check Type:  passive
Check Latency / Duration:  None / 0.0
Next Scheduled Active Check:  None
Last State Change:  2014-01-08 18:32:24
Last Update:  2014-02-06 10:07:06
 [root@cls12345n000 ~]# cscli monitor elements -S enclosures
cls12345n003-Enclosure-R1C1-21U  "FRU Fan Status":  OK  for 28d 15h 27m 54s  checked 2014-02-06 09:55:52  "All FRU's are operating normally"
cls12345n003-Enclosure-R1C1-21U  "FRU Power Supply Status":  OK  for 28d 15h 27m 54s  checked 2014-02-06 09:55:52  "All FRU's are operating normally"
cls12345n003-Enclosure-R1C1-21U  "FRU SBB Module Status":  OK  for 28d 15h 27m 54s  checked 2014-02-06 09:55:52  "All FRU's are operating normally"
cls12345n003-Enclosure-R1C1-21U  "Fan Statistics":  OK  for 15d 17h 19m  4s  checked 2014-02-06 09:58:06  "Summary: 4 Fan Sensors available. All Sensors readings are
within normal operating levels"
cls12345n003-Enclosure-R1C1-21U  "Power Statistics":  OK  for 15d 17h 19m  4s  checked 2014-02-06 09:58:06  "Summary: Total System Power 168W"
cls12345n003-Enclosure-R1C1-21U  "Thermal Statistics":  OK  for 15d 17h 19m  4s  checked 2014-02-06 09:58:06  "Summary: 6 Thermal Sensors available. All Sensors
readings are within normal operating levels"
cls12345n003-Enclosure-R1C1-21U  "Voltage Statistics":  OK  for 15d 17h 19m  4s  checked 2014-02-06 09:58:06  "Summary: 4 Voltage Sensors available. All Sensors
readings are within normal operating levels"
cls12345n005-Enclosure-R1C1-5U  "FRU Fan Status":  OK  for 28d 15h 27m 54s  checked 2014-02-06 09:55:52  "All FRU's are operating normally"
cls12345n005-Enclosure-R1C1-5U  "FRU Power Supply Status":  OK  for 20d 23h  0m 23s  checked 2014-02-06 09:55:52  "All FRU's are operating normally"
cls12345n005-Enclosure-R1C1-5U  "FRU SBB Module Status":  OK  for 28d 15h 27m 54s  checked 2014-02-06 09:55:52  "All FRU's are operating normally"
cls12345n005-Enclosure-R1C1-5U  "Fan Statistics":  OK  for 28d 15h 28m 40s  checked 2014-02-06 09:57:42  "Summary: 10 Fan Sensors available. All Sensors readings are
within normal operating levels"
cls12345n005-Enclosure-R1C1-5U  "Power Statistics":  OK  for 28d 15h 28m 40s  checked 2014-02-06 09:57:42  "Summary: Total System Power 1068W"
cls12345n005-Enclosure-R1C1-5U  "Thermal Statistics":  OK  for 28d 15h 28m 40s  checked 2014-02-06 09:57:42  "Summary: 13 Thermal Sensors available. All Sensors
readings are within normal operating levels"
cls12345n005-Enclosure-R1C1-5U  "Voltage Statistics":  OK  for 28d 15h 28m 40s  checked 2014-02-06 09:57:42  "Summary: 2 Voltage Sensors available. All Sensors
readings are within normal operating levels"
[root@cls12345n000 ~]#
[root@cls12345n000 ~]# cscli monitor nodes -n cls12345n004
cls12345n004:  UP  for 5d 17h 17m 26s  checked 2014-02-06 10:06:36  "PING OK - Packet loss = 0%, RTA = 0.17 ms"
[root@cls12345n000 ~]# cscli monitor elements  -n cls12345n004
cls12345n004  "Arrays and Disk Status":  OK  for 2d 14h 30m 42s  checked 2014-02-06 10:03:42  "All arrays are operating normally"
cls12345n004  "Current Load":  OK  for 28d 15h 34m 46s  checked 2014-02-06 10:03:42  "OK - load average: 0.01, 0.01, 0.01"
cls12345n004  "Current Users":  OK  for 28d 15h 34m 46s  checked 2014-02-06 10:03:42  "USERS OK - 0 users currently logged in"
cls12345n004  "Free Space":  OK  for 28d 15h 34m 46s  checked 2014-02-06 10:03:42  "DISK OK - free space: /tmp 15966 MB (99% inode=99%):"
cls12345n004  "Lustre Health":  OK  for 28d 15h 34m 46s  checked 2014-02-06 10:03:42  "OK:Lustre is ok"
cls12345n004  "Network statistics":  OK  for 28d 15h 34m 46s  checked 2014-02-06 10:03:42  "NET OK - (Rx/Tx) eth0=(16.9B/5.8B), ib0=(169.5B/60.9B), ib1=(0.0B/0.0B),
lo=(140.2B/140.2B), meth0=(16.9B/5.8B), meth1=(0.0B/0.0B), xyvnic0=(71.5B/75.2B)"
cls12345n004  "RAM usage":  OK  for 28d 15h 34m 46s  checked 2014-02-06 10:03:42  "OK - 12.9% (4203984 kB) used."
cls12345n004  "Swap Usage":  OK  for 28d 15h 34m 46s  checked 2014-02-06 10:03:42  "SWAP OK - 100% free (16386 MB out of 16386 MB)"
cls12345n004  "Total Processes":  OK  for 28d 15h 34m 46s  checked 2014-02-06 10:03:42  "PROCS OK: 1239 processes with STATE = RSZDT"
[root@cls12345n000 ~]# cscli monitor elements  -g oss
cls12345n004  "Arrays and Disk Status":  OK  for 2d 14h 31m 42s  checked 2014-02-06 10:08:43  "All arrays are operating normally"
cls12345n004  "Current Load":  OK  for 28d 15h 35m 46s  checked 2014-02-06 10:08:43  "OK - load average: 0.01, 0.01, 0.01"
cls12345n004  "Current Users":  OK  for 28d 15h 35m 46s  checked 2014-02-06 10:08:43  "USERS OK - 0 users currently logged in"
cls12345n004  "Free Space":  OK  for 28d 15h 35m 46s  checked 2014-02-06 10:08:43  "DISK OK - free space: /tmp 15966 MB (99% inode=99%):"
cls12345n004  "Lustre Health":  OK  for 28d 15h 35m 46s  checked 2014-02-06 10:08:43  "OK:Lustre is ok"
cls12345n004  "Network statistics":  OK  for 28d 15h 35m 46s  checked 2014-02-06 10:08:43  "NET OK - (Rx/Tx) eth0=(16.9B/5.8B), ib0=(169.5B/60.9B), ib1=(0.0B/0.0B),
lo=(140.3B/140.3B), meth0=(16.9B/5.8B), meth1=(0.0B/0.0B), xyvnic0=(71.5B/75.2B)"
cls12345n004  "RAM usage":  OK  for 28d 15h 35m 46s  checked 2014-02-06 10:08:43  "OK - 12.9% (4204568 kB) used."
cls12345n004  "Swap Usage":  OK  for 28d 15h 35m 46s  checked 2014-02-06 10:08:43  "SWAP OK - 100% free (16386 MB out of 16386 MB)"
cls12345n004  "Total Processes":  OK  for 28d 15h 35m 46s  checked 2014-02-06 10:08:43  "PROCS OK: 1239 processes with STATE = RSZDT"
cls12345n005  "Arrays and Disk Status":  OK  for 17d 14h 22m 41s  checked 2014-02-06 10:07:41  "All arrays are operating normally"
cls12345n005  "Current Load":  OK  for 28d 15h 38m  8s  checked 2014-02-06 10:07:41  "OK - load average: 0.01, 0.02, 0.02"
cls12345n005  "Current Users":  OK  for 28d 15h 38m  8s  checked 2014-02-06 10:07:41  "USERS OK - 0 users currently logged in"
cls12345n005  "Free Space":  OK  for 28d 15h 38m  8s  checked 2014-02-06 10:07:41  "DISK OK - free space: /tmp 15966 MB (99% inode=99%):"
cls12345n005  "Lustre Health":  OK  for 28d 15h 38m  7s  checked 2014-02-06 10:07:42  "OK:Lustre is ok"
cls12345n005  "Network statistics":  OK  for 28d 15h 38m  8s  checked 2014-02-06 10:07:41  "NET OK - (Rx/Tx) eth0=(25.8B/16.5B), ib0=(197.3B/64.8B), ib1=(0.0B/0.0B),
lo=(6.0B/6.0B), meth0=(25.8B/16.5B), meth1=(0.0B/0.0B), xyvnic0=(70.9B/76.1B)"
cls12345n005  "RAM usage":  OK  for 28d 15h 38m  8s  checked 2014-02-06 10:07:41  "OK - 12.8% (4192544 kB) used."
cls12345n005  "Swap Usage":  OK  for 28d 15h 38m  8s  checked 2014-02-06 10:07:41  "SWAP OK - 100% free (16386 MB out of 16386 MB)"
cls12345n005  "Total Processes":  OK  for 28d 15h 38m  8s  checked 2014-02-06 10:07:41  "PROCS OK: 1241 processes with STATE = RSZDT"
[root@cls12345n000 ~]#
[root@cls12345n000 ~]# cscli monitor elements -S arrays
cls12345n000  "Arrays and Disk Status":  OK  for 28d 15h 38m 14s  checked 2014-02-06 10:05:52  "All arrays are operating normally"
cls12345n001  "Arrays and Disk Status":  OK  for 28d 15h 39m 56s  checked 2014-02-06 10:08:07  "All arrays are operating normally"
cls12345n002  "Arrays and Disk Status":  OK  for 28d 15h 36m 38s  checked 2014-02-06 10:07:56  "All arrays are operating normally"
cls12345n003  "Arrays and Disk Status":  OK  for 28d 15h 36m 36s  checked 2014-02-06 10:06:24  "All arrays are operating normally"
cls12345n004  "Arrays and Disk Status":  OK  for 2d 14h 32m 35s  checked 2014-02-06 10:08:43  "All arrays are operating normally"
cls12345n005  "Arrays and Disk Status":  OK  for 17d 14h 23m 34s  checked 2014-02-06 10:07:41  "All arrays are operating normally"
[root@cls12345n000 ~]# cscli monitor elements -S arrays –v
```

Subset of output:

```
cls12345n000  "Arrays and Disk Status":  OK  for 28d 15h 39m 16s  checked 2014-02-06 10:10:52  "All arrays are operating normally
Array: md64, status: Ok, t10: disabled
Total number of disk slots available: 24
Total number of disks found: 24
slot:   2, wwn: 5000c50043b1e71f, cap:         450098159616, dev:   sdl, parts:        0, status: Hot Spare, t10: 11110100000
slot:  21, wwn: 5000c500479061af, cap:         450098159616, dev:   sdv, parts:        0, status: Hot Spare, t10: 11110100000
MD RAID to Lustre mapping
Array /dev/md/cls12345n003:md64 doesn't have associated WIB array
Degraded Array information:
All arrays are in clean state on node cls12345n000"
Performance Data:  None
Current Attempt:  1/3 (HARD state)
Check Type:  passive
Check Latency / Duration:  None / 0.0
Next Scheduled Active Check:  None
Last State Change:  2014-01-08 18:32:24
Last Update:  2014-02-06 10:11:36
------------------------------------------------------------------------
cls12345n001  "Arrays and Disk Status":  OK  for 28d 15h 40m 58s  checked 2014-02-06 10:08:07  "All arrays are operating normally
Array: md67, status: Ok, t10: disabled
Array: md127, status: Ok, t10: disabled
Total number of disk slots available: 24
Total number of disks found: 24
slot:   2, wwn: 5000c50043b1e71f, cap:         450098159616, dev:   sdv, parts:        0, status: Hot Spare, t10: 11110100000
slot:  21, wwn: 5000c500479061af, cap:         450098159616, dev:   sdc, parts:        0, status: Hot Spare, t10: 11110100000
MD RAID to Lustre mapping
Array /dev/md/cls12345n003:md67 doesn't have associated WIB array
Degraded Array information:
All arrays are in clean state on node cls12345n001"
Performance Data:  None
Current Attempt:  1/3 (HARD state)
Check Type:  passive
Check Latency / Duration:  None / 0.0
Next Scheduled Active Check:  None
Last State Change:  2014-01-08 18:30:42
Last Update:  2014-02-06 10:11:36
------------------------------------------------------------------------
cls12345n002  "Arrays and Disk Status":  OK  for 28d 15h 37m 40s  checked 2014-02-06 10:07:56  "All arrays are operating normally
Array: md65, status: Ok, t10: disabled
Total number of disk slots available: 24
Total number of disks found: 24
slot:   2, wwn: 5000c50043b1e71f, cap:         450098159616, dev:   sdv, parts:        0, status: Hot Spare, dev1: sdaj, t10: 11110100000
slot:  21, wwn: 5000c500479061af, cap:         450098159616, dev:   sdc, parts:        0, status: Hot Spare, dev1: sdat, t10: 11110100000
MD RAID to Lustre mapping
Array /dev/md/cls12345n003:md65 doesn't have associated WIB array
Target:     MGS
Degraded Array information:
All arrays are in clean state on node cls12345n002"
Performance Data:  None
Current Attempt:  1/3 (HARD state)
Check Type:  passive
Check Latency / Duration:  None / 0.0
Next Scheduled Active Check:  None
Last State Change:  2014-01-08 18:34:00
Last Update:  2014-02-06 10:11:36
------------------------------------------------------------------
[root@cls12345n000 ~]# cscli monitor elements -S disk
cls12345n000  "Arrays and Disk Status":  OK  for 28d 15h 43m 32s  checked 2014-02-06 10:10:52  "All arrays are operating normally"
cls12345n001  "Arrays and Disk Status":  OK  for 28d 15h 45m 14s  checked 2014-02-06 10:13:07  "All arrays are operating normally"
cls12345n002  "Arrays and Disk Status":  OK  for 28d 15h 41m 56s  checked 2014-02-06 10:12:56  "All arrays are operating normally"
cls12345n003  "Arrays and Disk Status":  OK  for 28d 15h 41m 54s  checked 2014-02-06 10:11:24  "All arrays are operating normally"
cls12345n004  "Arrays and Disk Status":  OK  for 2d 14h 37m 53s  checked 2014-02-06 10:13:42  "All arrays are operating normally"
cls12345n005  "Arrays and Disk Status":  OK  for 17d 14h 28m 52s  checked 2014-02-06 10:12:41  "All arrays are operating normally"
root@cls12345n000 ~]# cscli monitor elements -S fan
cls12345n003-Enclosure-R1C1-21U  "FRU Fan Status":  OK  for 28d 15h 43m 55s  checked 2014-02-06 10:10:52  "All FRU's are operating normally"
cls12345n003-Enclosure-R1C1-21U  "Fan Statistics":  OK  for 15d 17h 35m  5s  checked 2014-02-06 10:13:07  "Summary: 4 Fan Sensors available. All Sensors readings are 
within normal operating levels"
cls12345n005-Enclosure-R1C1-5U  "FRU Fan Status":  OK  for 28d 15h 43m 55s  checked 2014-02-06 10:10:52  "All FRU's are operating normally"
cls12345n005-Enclosure-R1C1-5U  "Fan Statistics":  OK  for 28d 15h 44m 41s  checked 2014-02-06 10:12:42  "Summary: 10 Fan Sensors available. All Sensors readings are 
within normal operating levels"
[root@cls12345n000 ~]# cscli monitor elements -S power
cls12345n003-Enclosure-R1C1-21U  "FRU Power Supply Status":  OK  for 28d 15h 44m  8s  checked 2014-02-06 10:15:53  "All FRU's are operating normally"
cls12345n003-Enclosure-R1C1-21U  "Power Statistics":  OK  for 15d 17h 35m 18s  checked 2014-02-06 10:16:25  "Summary: Total System Power 178W"
cls12345n005-Enclosure-R1C1-5U  "FRU Power Supply Status":  OK  for 20d 23h 16m 37s  checked 2014-02-06 10:15:53  "All FRU's are operating normally"
cls12345n005-Enclosure-R1C1-5U  "Power Statistics":  OK  for 28d 15h 44m 54s  checked 2014-02-06 10:12:42  "Summary: Total System Power 1061W"
[root@cls12345n000 ~]# cscli monitor elements -S sbb
cls12345n003-Enclosure-R1C1-21U  "FRU SBB Module Status":  OK  for 28d 15h 44m 23s  checked 2014-02-06 10:15:53  "All FRU's are operating normally"
cls12345n005-Enclosure-R1C1-5U  "FRU SBB Module Status":  OK  for 28d 15h 44m 23s  checked 2014-02-06 10:15:53  "All FRU's are operating normally"
[root@cls12345n000 ~]# cscli monitor elements -S volt
cls12345n003-Enclosure-R1C1-21U  "Voltage Statistics":  OK  for 15d 17h 35m 53s  checked 2014-02-06 10:16:24  "Summary: 4 Voltage Sensors available. All Sensors 
readings are within normal operating levels"
cls12345n005-Enclosure-R1C1-5U  "Voltage Statistics":  OK  for 28d 15h 45m 29s  checked 2014-02-06 10:12:42  "Summary: 2 Voltage Sensors available. All Sensors 
readings are within normal operating levels"
[root@cls12345n000 ~]# cscli monitor elements -S disk
cls12345n000  "Arrays and Disk Status":  OK  for 28d 15h 45m  0s  checked 2014-02-06 10:15:53  "All arrays are operating normally"
cls12345n001  "Arrays and Disk Status":  OK  for 28d 15h 46m 42s  checked 2014-02-06 10:13:07  "All arrays are operating normally"
cls12345n002  "Arrays and Disk Status":  OK  for 28d 15h 43m 24s  checked 2014-02-06 10:12:56  "All arrays are operating normally"
cls12345n003  "Arrays and Disk Status":  OK  for 28d 15h 43m 22s  checked 2014-02-06 10:16:24  "All arrays are operating normally"
cls12345n004  "Arrays and Disk Status":  OK  for 2d 14h 39m 21s  checked 2014-02-06 10:13:42  "All arrays are operating normally"
cls12345n005  "Arrays and Disk Status":  OK  for 17d 14h 30m 20s  checked 2014-02-06 10:12:41  "All arrays are operating normally"
[root@cls12345n000 ~]# cscli monitor elements -S arrays
cls12345n000  "Arrays and Disk Status":  OK  for 28d 15h 45m 10s  checked 2014-02-06 10:15:53  "All arrays are operating normally"
cls12345n001  "Arrays and Disk Status":  OK  for 28d 15h 46m 52s  checked 2014-02-06 10:13:07  "All arrays are operating normally"
cls12345n002  "Arrays and Disk Status":  OK  for 28d 15h 43m 34s  checked 2014-02-06 10:12:56  "All arrays are operating normally"
cls12345n003  "Arrays and Disk Status":  OK  for 28d 15h 43m 32s  checked 2014-02-06 10:16:24  "All arrays are operating normally"
cls12345n004  "Arrays and Disk Status":  OK  for 2d 14h 39m 31s  checked 2014-02-06 10:13:42  "All arrays are operating normally"
cls12345n005  "Arrays and Disk Status":  OK  for 17d 14h 30m 30s  checked 2014-02-06 10:12:41  "All arrays are operating normally"
[root@cls12345n000 ~]#
```

17.18 `netfilter_level` Command

Modes: Site configuration, Daily

The `netfilter_level` command manages the netfilter level on the Sonexion system.

> **IMPORTANT:** Exercise caution before using the `--force` parameter.

Synopsis

```
$ cscli netfilter_level [-h] [-s] [-l level] [--force]
```

where:

Optional Arguments	Description
`-h |--help`	Shows the help message and exits.
`-s |--show`	Shows the current netfilter level.
`-l level |--level level`	Sets the netfilter level (`off`, `lustre`, `on`).
`--force`	Forces the netfilter level to be set to off.

17.19 Network Setup Commands

The network setup commands manage network parameters for the Lustre file system. This command includes functions to show, set, apply, and reset Lustre network parameters. Users can also specify SSH TCP port settings.

Show Network Parameters

Modes: Site configuration, Guest

The `show_network_setup` command displays the Lustre network configuration. If the Lustre network is not yet configured, no parameters are shown.

Synopsis

```
$ cscli show_network_setup [-h] [-c cluster_name]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-c cluster_name |--cluster cluster_name`	Specifies the cluster name.

Set Network Parameters

Modes: Site configuration

The `set_network` command specifies new Lustre network parameters and adds them to the database.

Synopsis

For release 2.1.0:

```
$ cscli set_network [-h] -k netmask -r ipranges [-d dns] [-t ntp] [-c cluster_name]
```

Beginning with release 3.0:

```
$ cscli set_network [-h] -k netmask  [-d dns] [-t ntp]
```

Optional Arguments	Description
`-h  |--help`	Displays the help message and exits.
`-k` *`netmask`*`|--netmask` *`netmask`*	Specifies the network mask value of the IP address.
For release 2.1.0: `-r` *`ipranges`*`|--range` *`ipranges`*	Specifies the IP address range.
`-d` *`dns`* `|--dns` *`dns`*	Specifies the DNS server IP address (optional).
`ntp` `|--ntp` *`ntp`*	Specifies the NTP server's IP address (optional).
For release 2.1.0: `-c` *`cluster_name`* `|--cluster` *`cluster_name`*	Specifies the cluster name.

`-t`

Reset Network Parameters

Modes: Site configuration

The `reset_network_setup` command resets the Lustre network parameters by removing old values from the database and replacing them with default values.

> **IMPORTANT:** Exercise caution before using the `-y` or `|--yes` parameter.

Synopsis

```
$ cscli reset_network_setup [-h] [-y] [-c cluster_name]
```

Optional Arguments	Description
`-h  |--help`	Displays the help message and exits.
`-y  |--yes`	Confirms the action to reset the network parameters.
`-c` *`cluster_name`*`|--cluster` *`cluster_name`*	Specifies the cluster name.

`Apply Network Setup` Command

Modes: Site configuration

(Deprecated beginning with release 3.0.0; see `lustre_network apply` subcommand.

The `apply_network_setup` command applies new Lustre network parameters to the database.

> **IMPORTANT:** Exercise caution before using the `--y` or `|--yes` parameter.

Synopsis

```
$ cscli apply_network_setup [-h] [--yes] [-c cluster_name]
```

Optional Arguments	Description
`-h|--help`	Displays the help message and exits.
`|--yes`	Confirms the action that network setup parameters were applied.

Optional Arguments	Description
-c *cluster_name*|--cluster *cluster_name*	Specifies the cluster name.

Specify SSH TCP Port Settings

Modes: Site configuration

The ssh_port commands are used to specify the SSH TCP port settings. *(Introduced with 2.1.0.)*

Synopsis

```
$ cscli ssh_port  [-h] [--yes] {status,apply,clear,set}
```

Positional Arguments	Description
clear	Disable SSH port redirection, return SSH to default port (22.)
set	Assign new port to SSH, leave existing one as a backup
status	Show current SSH port status.
apply	Stop listening to backup port

Optional Arguments	Description
-h |--help	Displays the help message and exits.

ssh_port set Subcommand

The ssh_port set command is a subcommand of the ssh_port command. *(Introduced with 2.1.0.)*

Synopsis

```
$ cscli ssh_port set [-h] [-p 22022]
```

where:

Optional Arguments	Description
-p 22022 |--port 22022	New SSH port to listen to.
-h |--help	Displays the help message and exits.

17.20 Node Control Commands

The node control commands are used to control individual Lustre nodes (MDS, MGS, and OSS) in a clustered file system. The commands include functions to mount and unmount the Lustre nodes, show nodes in the file system.

Additional functions include powering nodes on and off, managing node failover and failback, and managing node auto-discovery.

`autodiscovery_mode` Command

Modes: Site configuration, Daily

The `autodiscovery_mode` command manages node auto-discovery in the Sonexion system.

Synopsis

```
$ cscli autodiscovery_mode [-h] [-s] [--mode {enabled,disabled}]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-s |--status`	Indicates the status of the auto-discovery mode.
`--mode {enabled,disabled}`	Switches to the specified mode. Enables or disables the auto-discovery mode.

Node `failback` and `failover` Commands

Modes: Daily

The `failover` and `failback` commands manage node failback and failover in the Sonexion system.

Synopsis

```
$ cscli failback [-h] (-F filter_sid | -n node_spec) |-c cluster_name |--cluster
cluster_name
$ cscli failover [-h] (-F filter_sid | -n node_spec) |-c cluster_name |--cluster
cluster_name
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-f` *`filter_sid`* `|--filter` *`filter_sid`*	The filter identifier for the specified node. Failover/ failback actions run on the nodes by filtering this filter.
`-n` *`node_spec`* `|--nodes` *`node_spec`*	Specifies the nodes on which the failover/failback operations are performed. Node hostnames should be passed in pdsh style. If this parameter is passed, the `--filter` parameter is ignored.
`-c` *`cluster_name`*`|--cluster` *`cluster_name`*	This parameter is deprecated. It is supported only for backward compatibility.

Lustre Target `mount` and `unmount` Commands

Modes: Site configuration, Daily

The `mount` and `unmount` commands control file system access to the Lustre targets (MDS, MGS, and OSS). The mount action enables file system access to the node. The unmount action disables file system access to the node.

If one or more nodes are specified, the mount/unmount action is performed only on the selected nodes in the file system. If no server nodes are specified, then the mount/unmount action is performed on all server nodes in the file system.

> **NOTE:** Exercise caution before using the `--force` parameter.

Synopsis

```
$ cscli mount [-h] -f fs_name [-n node_spec] |-c cluster_name |--cluster
cluster_name
$ cscli unmount [-h] -f fs_name [-n node_spec] |-c cluster_name |--cluster
cluster_name [-evict] [-force]
```

Optional Arguments	Description
`-h  |--help`	Displays the help message and exits.
`-f fs_name |--fs-name=fs_name`	Specifies the name of the file system.
`-n node_spec |--nodes=node_spec`	Specifies the node(s) on which the mount/unmount action is performed. Node hostnames should be passed in pdsh style.
`-c cluster_name|--cluster cluster_name`	This parameter is deprecated. It is supported only for backward compatibility.
`|--evict`	This parameter evicts clients before they unmount.
`|--force`	This parameter is the force mode to evict Lustre clients.

Node `power_manage` Command

Modes: Daily

The `power_manage` command manages the power on the Sonexion system. These commands power-cycle nodes on and off and also control HA resource hand-offs.

> **IMPORTANT:** Exercise caution before using the `--force` parameter.

Synopsis

```
$ cscli power_manage [-h] (--filter filter_sid
| -n node_spec) (--power-on | --power-off | --reboot | --cycle | --reset
| --hand-over) [--force] -c cluster_name, --cluster cluster_name
```

Optional Arguments	Description
`-h  |--help`	Displays the help message and exits.
`-f filter_sid |--filter filter_sid`	The filter identifier for the specified node. Failover and failback actions run on the nodes by filtering this filter. If `--filter` is specified, then `--nodes` is ignored.
`-n node_spec`	Specifies the nodes on which failover/failback operations are performed. Node hostnames should be passed in pdsh style.
`--power-on`	Powers on the specified nodes.

Optional Arguments	Description
`--power-off`	Powers off the specified nodes.
`--reboot`	Reboots the specified nodes.
`--cycle`	Power-cycles the specified nodes.
`--reset`	Resets the specified nodes.
`--hand-over`	Hands over resources.
`--force`	An optional flag that indicates the node operation should be performed in force mode; should only be used with `--power-off`.
`-c` *`cluster_name`*`|--cluster` *`cluster_name`*	*This parameter is deprecated.* It is supported only for backward compatibility.

`remove_unit` Command

The `remove_unit` command is used to remove units from the cluster.

Synopsis

```
$ cscli remove_unit [-h] [-n NODES] [-p DUMP_PATH]
```

where:

Option	Description

`-n NODES`
`|--node NODES-`
	pdsh-style nodes hostnames that should be removed.

`-p DUMP_PATH`
`|--dump-path DUMP_PATH`
	Path to file for dumping removed nodes information.

`-h |--help`	Displays the help message and exits.

`show_nodes` Command

Modes: Guest, Daily

The `show_nodes` command displays information about specified system nodes.

Synopsis

```
$ cscli show_nodes [-h] [-F filter_sid] [-r] |-c cluster_name |--cluster
cluster_name
```

where:

Option	Description
`-h |--help`	Displays the help message and exits.
`-F` *`filter_sid`* `|--filter` *`filter_sid`*	Specifies the node filter.

Option	Description
`-r |--refresh`	Specifies the refresh mode (press **q** for quit).
`-c cluster_name|--cluster cluster_name`	*This parameter is deprecated.* It is supported only for backward compatibility.

17.21 Node Position and Management Commands

The rack position commands manage the location of components (hosting system nodes) in a Sonexion rack. The SMU hosts the primary and secondary MGMT, MGS and MDS nodes. Each SSU hosts OSS nodes (two OSSes per SSU).

get_rack_position Command

Modes: Daily

The get_rack_position command indicates the location of server nodes in a Sonexion rack.

Synopsis

```
$ cscli get_rack_position [-h] [-l] [-r racks] [-a] [--yaml]
```

Option	Description
`-h |--help`	Displays the help message and exits.
`--yaml`	Prints node rack position information in YAML file format.
`-l |--list`	Displays the names of all the racks.
`-r racks|--racks racks`	Displays the hostnames in all positions within the specified racks. *racks* is a comma-separated list of rack names.
`-a |--all`	Displays information about all racks.

set_rack_position Command

Modes: Daily

Deprecated in release 3.0.0. See the rack move command.

The set_rack_position command sets the location of server nodes in the Sonexion rack. It changes the position of one node in the rack (or moves the node to another rack).

IMPORTANT: Exercise caution before using the --force parameter.

Synopsis

```
$ cscli set_rack_position [-h] (-y yaml_path | -r rack_name -n node_name -p
position)
```

Optional Arguments	Description
-h |--help	Displays the help message and exits.
--force	Creates a new rack if not found.
-y *yaml path* |--yaml *yaml_path*	Loads rack position information from *yaml* file format.
-r *rack_name* |--rack *rack_name*	Specifies the rack containing the node(s). This can be set manually.
-n *node_spec* |--node *node_spec*	Specifies the node(s) hostname. This can be set manually.
-p *position* |--position=*position*	Specifies the node position in rack units. This can be set manually.

show_update_versions Command

Modes: Daily

The show_update_versions command lists software versions available in the Sonexion Management (MGMT) Server repository.

Synopsis

```
$ cscli show_update_versions [-h]
```

Option	Description
-h |--help	Displays the help message and exits.

show_version_nodes Command

Modes: Daily

The show_version_nodes command lists all system nodes at the specified software version.

Synopsis

```
$ cscli show_version_nodes [-h] [-q] -v sw_version
```

Option	Description
-h |--help	Displays the help message and exits.
-q |--query	Controls the format of the command output. If this flag is specified, nodes in output should display in genders style. For example, snx11000n[002-005,097-098].
-v *sw_version*|--version *sw_version*	Specifies the Sonexion software version.

show_node_versions Command

Modes: Daily

The show_node_versions command displays the Sonexion software version running on specified nodes.

Synopsis

```
$ cscli show_node_versions [-h] [-q] [-n node_spec] [-g genders_query]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-q, --query`	Controls output format. If this flag is specified, nodes in the output should be in genders style. For example, `snx11000n[002-005,097-098]`
`-n` *node_spec*, `--nodes` *node_spec*	Specifies nodes to indicate the Sonexion software version.
`-g` *genders_query*	Specifies a gender's style query.

17.22 RAID Management Commands

The `raid` command is used to manage the system's RAID configuration.

Synopsis

```
$ cscli raid [-h] {disk_fail,speed,check,show}
```

where:

Positional Arguments	Description
`show`	Show raid configuration.
`check`	Show settings related to RAID and disk checks, and whether checks are running.
`speed`	Show raid configuration.
`disk_fail`	Settings for whether a disk will be forced to a "failed" state.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

`raid show` Subcommand

The `raid show` command is a subcommand of the `raid` command.

Synopsis

```
$ cscli raid show [-h] [-n nodespec]
```

where:

Optional Arguments	Description
`-n` *nodespec* `|--node` *nodespec*	Node specification.
`-h |--help`	Displays the help message and exits.

`raid check` Subcommand

Modes: Daily

The `raid check` command is a subcommand of the `raid` command that enables RAID check on RAID devices.

Synopsis

```
$ cscli raid check [-h] {schedule,disks,abort,limit,show,now} options
```

Positional Arguments	Description
`show`	Show RAID configuration.
`now`	Perform an immediate RAID check.
`abort`	Cancel a running RAID check.
`schedule`	Enable, disable, or change schedule of RAID checks.
`disks`	Set interval in seconds for disk checks.
`limit`	Set RAID check concurrency.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

`raid check now` Subcommand

Modes: Daily

The `raid check now` command is a second-level subcommand of the `raid` command. Use the command to perform an immediate RAID check.

Synopsis

```
$ cscli raid check now [-h] [-n nodespec]
```

where:

Positional Arguments	Description
`-n` *nodespec* `|--node` *nodespec*	Look through passed hostnames elements. pdsh-style nodes hostnames.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

`raid check abort` Subcommand

Modes: Daily

The `raid check abort` command is a second-level subcommand of the `raid` command. Use the command to cancel a RAID check operation that is already running.

Synopsis

```
$ cscli raid check abort [-h] [-n nodespec]
```

where:

Positional Arguments	Description
`-n` *nodespec* `|--node` *nodespec*	Look through passed hostnames elements. pdsh-style nodes hostnames.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

`raid check schedule` Subcommand

Modes: Daily

The `raid check schedule` command is a second-level subcommand of the `raid` command. Use this command to change the schedule for running RAID checks.

Synopsis

```
$ cscli raid check schedule [-h] disks,abort,limit,show,now} (--enable | --
disable)
                                       [--at atime] [-n nodespec
```

where:

Positional Arguments	Description
`--enable`	Enable cron job for raid check.
`--disable`	Disable cron job for raid check.
`--at` *atime*	Time when command should be run (see "man at").
`-n` *nodespec* `|--node` *nodespec*	Look through passed hostnames elements. pdsh-style nodes hostnames.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

raid check disks Subcommand

Modes: Daily

The raid check disks command is a second-level subcommand of the raid command. Use this command to set the interval for disk checks.

Synopsis

```
$ cscli raid check disks [-h] (--interval interval | --reset)
[-n nodespec]
```

where:

Positional Arguments	Description
--interval *interval*	Configure interval on which DWD will re-check drives. This parameter expects integer (number) as an argument. Interval is set in seconds and defaults to 24 hours (86400 seconds).
--reset	Reset interval to default 24 hours (86400 seconds) on which DWD will re-check drives.
-n *nodespec* |--node *nodespec*	Look through passed hostnames elements. pdsh-style nodes hostnames.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

raid check limit Subcommand

Modes: Daily

The raid check limit command is a second-level subcommand of the raid command. Use this command to set RAID check concurrency.

Synopsis

```
$ cscli raid check limit [-h] [-n nodespec] (--concurrent concurrency | --reset)
```

where:

Positional Arguments	Description
--concurrent *concurrency* |-c *concurrency*	Limit number of simultaneous RAID checks to NUM on the specified nodes. If no nodes are specified, apply the setting to all nodes.
--reset	Reset the limit to the default value, 8.
-n *nodespec* |--node *nodespec*	Look through passed hostnames elements. pdsh-style nodes hostnames.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

`raid speed` Subcommand

Modes: Daily

The `raid speed` command is a subcommand of the `raid` command.

Synopsis

```
$ cscli raid speed [-h] {multiple,first,rebalance,check,show}
```

where:

Positional Arguments	Description
`show`	Show RAID configuration.
`first`	This speed (in KiB/s) controls the rebuild rate for a RAID-6 array after a single drive loss.
`rebalance`	Use this speed in KB/s to rebalance data, parity, and spare space after GridRAID reconstruction for a one-drive failure. For RAID-6, this setting is ignored.
`multiple`	When multiple disks fail in an array, increase to this speed in KB/s. For GridRAID, this applies both to reconstruction and rebalancing. For RAID-6, this applies to rebuild.
`check`	Use this speed in KB/s to perform periodic data integrity checks on arrays. See "cscli raid check" for scheduling options.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

`raid speed first` Subcommand

Modes: Daily

The `raid speed first` command is a second-level subcommand of the `raid` command. Use this command to specify the speed (in KiB/s) to control the rebuild rate for a RAID-6 array after a single drive loss.

Synopsis

```
$ cscli raid speed first [-h] [--set-min single_min_rate]
                                             [--set-max single_max_rate] [--reset]
                                             [-n nodespec | -a]
```

where:

Positional Arguments	Description
`--set-min` *`single_min_rate`*	After a single drive failure, use this min rate to recover array redundancy. For RAID-6, this is the min rebuild rate. For GridRAID, this is the min reconstruction rate.
`--set-max` *`single_max_rate`*	After a single drive failure, use this max rate to recover array redundancy. For RAID-6, this is the max rebuild rate. For GridRAID, this is the max reconstruction rate.
`--reset`	

`-n` *`nodespec`*
`|--node`*`nodespec`*
	Look through passed hostnames elements. pdsh-style nodes hostnames.

`-a` `|--all`	All nodes.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

raid speed rebalance Subcommand

Modes: Daily

The `raid speed rebalance` command is a second-level subcommand of the `raid` command. Use this command to specify the speed in KB/s to rebalance data, parity, and spare space after GridRAID reconstruction for a one-drive failure. For RAID-6, this setting is ignored.

Synopsis

```
$ cscli raid speed rebalance [-h] [--set-min rebalance_min_rate]
                                                  [--set-max rebalance_max_rate]
[--reset]
                                                  [-n nodespec | -a]
```

where:

Positional Arguments	Description
`--set-min` *`rebalance_min_rate`*	For GridRAID, use this min rate to rebalance data, parity, and spare space after reconstruction is completed for a single drive failure. For RAID-6, this parameter is ignored.
`--set-max` *`rebalance_max_rate`*	For GridRAID, use this max rate to rebalance data, parity, and spare space after reconstruction is completed for a single drive failure. For RAID-6, this parameter is ignored.
`--reset`	

`-n` *`nodespec`*
`|--node` *`nodespec`*
	Look through passed hostnames elements. pdsh-style nodes hostnames.

Positional Arguments	Description
-a |--all	All nodes.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

raid speed multiple Subcommand

Modes: Daily

The raid speed multiple command is a second-level subcommand of the raid command. Use this command to specify an increase in speed, in KB/s, to use when multiple disks fail in an array. For GridRAID, this applies both to reconstruction and rebalancing. For RAID-6, this applies to rebuild.

Synopsis

```
$ cscli raid speed multiple [-h] [--set-min multiple_min_rate]
                                               [--set-max multiple_max_rate] [--
reset]
                                               [-n nodespec | -a]
```

where:

Positional Arguments	Description
--set-min *multiple_min_rate*	After multiple drive failure, increase to this min rate. For RAID-6, this is the min rebuild rate. For GridRAID, this is the min reconstruction rate and the min rebalance rate.
--set-max *multiple_max_rate*	After multiple drive failure, increase to this max rate. For RAID-6, this is the max rebuild rate. For GridRAID, this is the max reconstruction rate and the max rebalance rate.
--reset	
-n *nodespec* |--node *nodespec*	Look through passed hostnames elements. pdsh-style nodes hostnames.
-a |--all	All nodes.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

raid speed check Subcommand

Modes: Daily

The `raid speed check` command is a second-level subcommand of the `raid` command. Use this command to specify the speed, in KB/s, to perform periodic data integrity checks on arrays. See `cscli raid check` for scheduling options.

Synopsis

```
$ cscli raid speed check [-h] [--set-min check_min_rate]
                                              [--set-max check_max_rate] [--reset]
                                              [-n nodespec | -a]
```

where:

Positional Arguments	Description
`--set-min` *`check_min_rate`*	Min speed in KB/s to perform periodic data integrity checks on array.
`--set-max` *`check_max_rate`*	Max speed in KB/s to perform periodic data integrity checks on array.
`--reset`	

`-n` *`nodespec`*
`|--node` *`nodespec`*
	Look through passed hostnames elements. pdsh-style nodes hostnames.

`-a` `|--all`	All nodes.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

`raid disk_fail` Subcommand

Modes: Daily

The `raid disk_fail` command is a subcommand of the `raid` command.

Synopsis

```
$ cscli raid disk_fail [-h]
{read_errors_raid6,offline,scsi_aborts,read_errors,show}
```

where:

Positional Arguments	Description
`show`	Show all settings related to forced disk failure.
`read errors`	Set the number of read errors that will cause a disk to be force-failed. Applies to GridRAID and/or RAID-6.
`read_errors_raid6`	Set the number of read errors that will cause a disk to be force-failed. Overrides "read_errors" for RAID-6.
`scsi_aborts`	Set the number of SCSI Task Aborts that will cause a disk to be force-failed.

Positional Arguments	Description
`offline`	Enable or disable taking a disk drive offline (power-off) for either a predictive SMART failure or too many SCSI Task Aborts.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

`raid disk_fail read_errors` Subcommand

Modes: Daily

The `raid disk_fail read_errors` command is a second-level subcommand of the `raid` command. Use the command to set the number of read errors that will cause a disk to be force-failed. Applies to GridRAID and/or RAID-6.

Synopsis

```
$ cscli raid disk_fail read_errors [-h] (--set max_read_errs | --reset)
                                         [-n nodespec | -a]
```

where:

Positional Arguments	Description
`--set` *max_read_errs*	Set max number of read errors threshold.
`--reset`	Reset max number of read errors to default value.
`-n` *nodespec* `|--node` *nodespec*	Look through passed node names. pdsh-style nodes hostnames.
`-a` `|--all`	All nodes.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

`raid disk_fail read_errors_raid6` Subcommand

Modes: Daily

The `raid disk_fail read_errors_raid6` command is a second-level subcommand of the `raid` command. Use the command to set the number of read errors that will cause a disk to be force-failed. Overrides "read_errors" for RAID-6.

Synopsis

```
$ cscli raid disk_fail read_errors_raid6 [-h]
                                         [--set max_read_errs_r6 | --reset]
                                         [-n nodespec | -a]
```

where:

Positional Arguments	Description
`--set` *max_read_errs_r6*	Set max number of read errors threshold.
`--reset`	Reset max number of read errors to default value.
`-n` *nodespec* `|--node` *nodespec*	Look through passed node names. pdsh-style nodes hostnames.
`-a` `|--all`	All nodes.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

raid disk_fail scsi_aborts Subcommand

Modes: Daily

The `raid disk_fail scsi_aborts` command is a second-level subcommand of the `raid` command. Use this command to set the number of SCSI Task Aborts that will cause a disk to be force-failed.

Synopsis

```
$ cscli raid disk_fail scsi_aborts [-h]
                                    (--none | --auto | --set max_aborts
| --reset)
                                    [-n nodespec | -a]
```

where:

Positional Arguments	Description
`--set` *max_aborts*	Set max number of aborts that will cause a disk to be force-failed.
`--reset`	Reset max number of aborts to defaults.
`-n` *nodespec* `|--node` *nodespec*	Look through passed node names. pdsh-style nodes hostnames.
`-a` `|--all`	All nodes.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

raid disk_fail offline Subcommand

Modes: Daily

The `raid disk_fail offline` command is a second-level subcommand of the `raid` command. Use the command to enable or disable taking a disk drive offline (power-off) for either a predictive SMART failure or too many SCSI Task Aborts.

Synopsis

```
$ cscli raid disk_fail offline [-h] (--enable | --disable | --reset)
                                    [-n nodespec | -a]
```

where:

Positional Arguments	Description
`--enable`	Enable taking a disk drive offline.
`--disable`	Disable taking a disk drive offline.
`--reset`	Reset (enable as its default) taking a disk drive offline.

`-n` *`nodespec`*
`|--node` *`nodespec`*
	Look through passed node names. pdsh-style nodes hostnames.

`-a` `|--all`	All nodes.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

17.23 Rack Management Commands

Modes: Daily

The `rack` command has options to list, show, create, delete, move and rename racks.

Synopsis

```
$ cscli rack [-h] {rename,show,create,move,list,delete} ...
```

where:

Positional Arguments	Description
`list`	Lists racks
`show`	Shows racks
`create`	Creates racks
`delete`	Deletes racks
`move`	Moves racks

Positional Arguments	Description
rename	Renames racks

Optional Arguments	Description
-h |--help	Displays the help message and exits.

17.24 service_console Commands

Modes: Daily

The service_console command lets a site administrator manage the system service console and block or allow its use for a service technician.

The service console is an alternate system management interface that is limited to maintenance functions, and is for use by authorized service personnel only. Even though it bypasses the normal administrative login, the service console cannot be used to access the root account, or to access data on the Lustre file system.

While it is not necessary to disable the service console for any system that is physically secure, administrators in high-security environments might prefer to disable it. In that case, they will need to re-enable the service console each and every time a service technician arrives on site to perform system maintenance, including replacement of failed disk drives and other parts.

Configuration consists of two main areas

- Configuring the SMTP relay, this allows the system to send emails to users and service emails.
- Configuring user email addresses to allow service email to go to system users.

This following commands and available subcommands are used to manage the service console.

Synopsis

```
$ cscli service_console [-h] {enable,disable,show,notifications,configure}
```

where:

Subcommands	Description
enable	Enables the service console
disable	Disables the service console
show	Show service console enabled status
notifications	View RAS notifications
configure	Configure RAS notifications

Optional Arguments	Description
-h |--help	Displays the help message and exits

Examples:

Enable the service console.

```
$ cscli service_console enable
Service Console enabled.
```

Disable the Service console.

```
$ cscli service_console disable
Service Console disabled.
```

Show the status of the service console (enabled).

```

$ cscli service_console show
Service console enabled.
```

Show service console notifications (service events).

```
$ cscli service_console notifications show
Current outstanding service call events:

Item 1 : "Disk drive needs replacement"

Service Code: 002005001
Time the event was first detected: Wed, 17 Jun 2015 17:18:14 EDT

    Details of failed component
    Disk Serial #: S0M122HN0000B40298QD
    Disk Model: ST600MM0006 Drive
    Manufacturer: SEAGATE
    Firmware version of drive at time of failure: XLGE T10     Enabled: Yes

Location of failed component

    Rack Name: Rack1
    Enclosure Model: 2U24
    Enclosure Location: 36U
    Disk located in slot: 15

Item 2: "Power supply issue detected"

Service Code: 002005003
Time the event was first detected: Wed, 02 Dec 2015 12:58:57 PST

Details of failed component

    Power supply chassis type: OneStor Power One 764W_AC_PCM (Original Silver DFM).
    Power supply part number: 0945768-10
    Power supply product version: 0314
    Power supply serial number: PMW0945768J1BDY

Location of failed component

    Rack Name: R1C1
    Enclosure Model: 2U24
    Enclosure Location: 24U
    Power supply located in bay: 1
```

Show service console configuration options.

```
$ cscli service_console configure
Commands:
    smtp                Configure SMTP settings
    snmp                Configure SNMP settings
    email               Configure email for sending RAS notifications
    system              Configure system settings
    rest_api            Configure REST API settings
```

`notifications` Subcommand

The `notifications` command is a subcommand of the `service_console` command, and is used to view notifications.

Synopsis

```
$ cscli service_console notifications -h {show}
```

Subcommands	Description
show	Show outstanding notifications.

Optional Arguments	Description
-h |--help	Displays the help message and exits.

Service Console Configuration Commands

These commands are used to configure the SMTP relay, SNMP, remote support, email addresses for RAS notifications, system and REST API settings.

Synopsis

```
$ cscli service_console configure {smtp,snmp,remote_support,email,system,rest_api}
```

Subcommands	Description
smtp	Configures SMTP settings
snmp	Configures SNMP settings
remote_support	Configure remote support RAS notifications
email	Configure email for sending RAS notifications
system	Configure system settings
rest_api	Configure REST API settings

Optional Arguments	Description
-h |--help	Displays the help message and exits

smtp Commands

The smtp command is a second-level subcommand of the service_console command, and is used to configure SMTP.

Synopsis

```
$ cscli service_console configure smtp -h -H {relay,show}
$
```

Subcommands	Description
relay	Configures SMTP relay for sending RAS notifications
show	Displays current SMTP configuration

Optional Arguments	Description
-H |--host	SMTP host (required)
-h |--help	Displays the help message and exits

relay Command

The relay command is the third-level subcommand of the service_console command, and is used to configure SMTP relay.

Synopsis

```
$ cscli service_console configure smtp relay [-h] -H smtp_host -P smtp_port
```

Subcommands	Description
-P *smtp_port* |--port *smtp_port*	SMTP port [default: 25]
-H *smtp_host* |--host *smtp_host*	SMTP host (required)
-h |--help	Displays the help message and exits

Examples:

Configure the SMTP relay.

```
$ cscli service_console configure smtp relay -H
mailrelayus.xyus.xyratex.com
OK.
```

Show the current SMTP configuration.

```
$ $ cscli service_console configure smtp show
SMTP relay: mailrelayus.xyus.xyratex.com:25
```

snmp Command

The snmp command is a second-level subcommand of the service_console command, and is used to configure SNMP settings

Synopsis

```
$ cscli service_console configure snmp -h {enable,disable,show}
```

Subcommands	Description
enable	Enables SNMP monitoring
disable	Disables SNMP monitoring
show	Shows current SNMP settings

Optional Arguments	Description
-h |--help	Displays the help message and exits

remote_support Commands

The remote_support command is a second-level subcommand of the service_console command, and is used to configure SMTP remote support.

Synopsis

```
$ cscli service_console configure remote_support -h {enable,disable,show}
```

Subcommands	Description
enable	Enables remote support
disable	Disables remote support
show	Shows current remote support status

Optional Arguments	Description
-h |--help	Displays the help message and exits

Examples:

Disable remote support.

```
$ cscli service_console configure remote_support disable

Remote Support disabled.
```

Display remote support status.

```
$ cscli service_console configure remote_support show
Remote Support enabled.
```

Enable remote support.

```
$ cscli service_console configure remote_support enable
Remote Support enabled.
```

Email Notification Commands

This command configures email notifications, including enabling or disabling notification functionality, adds or deletes email addresses to the user list, shows user list entries and sends test email to determine if user notifications can be sent successfully.

Synopsis

```
$ cscli service_console configure email [-h]
{show,add,delete,enable,diable,send_test_email}
```

Subcommands	Description
show	Shows all notification email addresses
add	Adds a notification email address. Also specify the email address to be added (required): -A *email_address* |--address *email_address*
delete	Deletes a notification email address Also specify the email address to be added (required): -A *email_address* |--address *email_address*
enable	Enables email notifications
disable	Disables email notifications
send_test_email	Sends test email to all enabled services

Optional Arguments	Description
-h |--help	Displays the help message and exits

Examples:

Show email addresses configured to receive RAS notifications (no configured addresses).

```
$  cscli service_console configure email show
No email addresses are configured to receive service notifications.
```

Show email addresses configured to receive RAS notifications (several configured addresses).

```
$  $ cscli service_console configure email show
The following email addresses are configured to receive service notifications:
sam_jones@xyzcorp.com
amy_cooper@xyzcorp.com
```

Add an email address to the user list (success).

```
$ cscli service_console configure email add -A sam_jones@xyzcorp.com
Successfully added 'sam_jones@xyzcorp.com' to receive notification emails.
```

Add an email address to the user list (fails because it is currently associated with the user list).

```
$ cscli service_console configure email add -A sam_jones@xyzcorp.com
Unable to add: 'sam_jones@xyzcorp.com' is already configured for receiving
notification emails.
```

Delete an email address from the user list (success).

```
$ cscli service_console configure email delete -A sam_jones@xyzcorp.com
Successfully removed 'sam_jones@xyzcorp.com' from receiving notification emails.
```

Delete an email address from the user list (fails because it is not associated with the user list).

```
$ cscli service_console configure email delete -A sam_jones@xyzcorp.com
Unable to delete: 'sam_jones@xyzcorp.com' is not configured for receiving
notification emails.
```

Enable email notifications.

```
$ cscli service_console configure email enable
OK.
```

Disable email notifications.

```
$ cscli service_console configure email disable
OK.
```

`system` Command

The `system` command is a second-level subcommand of the `service_console` command, and is used to configure system settings.

Synopsis

```
$ cscli service_console configure system [-h] {identifier,show}
```

Subcommands	Description
`identifier`	Sets the system identifier
`show`	Shows the current system identifier (name)

Optional Arguments	Description
`-h |--help`	Displays the help message and exits

Examples:

Show the current system identifier (name specified).

```
$ cscli service_console configure system show
System settings:
    System identifier name: snx-002
```

Show the current system identifier (no name set).

```
$ cscli service_console configure system show
```

identifier Command

The `identifier` command is a third-level subcommand of the `service_console` command, and is used to set the system identifier name.

Synopsis

```
$ cscli service_console configure system identifier -h -n id_name
```

Usage:

```
$ cscli service_console [options] configure [options]
system [options] identifier [options]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits
`-n` *`id_name`* `|--name=`*`id_name`*	Name (required)

Example:

Set the system identifier.

```
$ cscli service_console configure system identifier –name=snx-002
System identifier name has been set to: snx-002
```

rest_api Command

The `rest_api` command is a second-level subcommand of the `service_console` command, and is used to configure REST API settings.

Synopsis

```
$ cscli service_console configure rest_api -h {enable,disable,show}
```

Subcommands	Description
`enable`	Enables the REST API
`disable`	Disables the REST API
`show`	Shows current status of the REST API

Optional Arguments	Description
`-h |--help`	Displays the help message and exits

17.24.1 `service_console configure` Command

Modes: Daily

The `service_console configure` command includes subcommands to configure Service Command help.

Synopsis

```
$ cscli service_console configure -h
```

Usage:

```
cscli service_console [options] configure [options]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits

Subcommands	Description
`smtp`	Configures SMTP settings
`snmp`	*(Added for release 2.1.0.)* Configure SNMP settings
`remote_support`	*(Added for release 2.1.0.)* Configure remote support RAS notifications
`email`	Configure e-mail for sending RAS notifications
`system`	*(Added for release 2.1.0.)* Configure system settings
`rest_api`	*(Added for release 2.1.0.)* Configure REST API settings

`configure smtp` Subcommand

The `configure smtp` command is a subcommand of the `configure` command, and configures SMTP.

Synopsis

```
$ cscli service_console configure smtp -h
```

Usage:

```
cscli service_console [options] configure [options] smtp [options]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits

Subcommands	Description
`relay`	Configures SMTP relay for sending RAS notifications
`show`	Displays the SMTP configuration

`configure smtp_relay` Subcommand

The `configure smtp_relay` command is the second-level subcommand of the `smtp` command, and configures SMTP relay.

Synopsis

```
$ cscli service_console configure smtp relay -h
```

Usage:

```
cscli service_console [options] configure [options]
smtp [options] relay [options]
```

Optional Arguments	Description
`-H |--host`	SMTP host (required)
`-P |--port`	SMTP port [default: 25]
`-h |--help`	Displays the help message and exits

`configure email` Subcommand

The `configure email` command is a second-level subcommand of the `configure` command, and is used to configure email.

Synopsis

```
$ cscli service_console configure email -h
```

Usage:

```
cscli service_console [options] configure [options] email [options]
```

where:

Optional Arguments	Description
`-h |--help`	Displays the help message and exits

Subcommands	Description
`show`	Shows all notification e-mail addresses
`add`	Adds a notification e-mail address
`delete`	Deletes a notification e-mail address
`enable`	Enables e-mail notifications
`disable`	Disables e-mail notifications
`send_test_email`	Sends test email to all enabled services

`configure email_add` Subcommand

This command is a second-level subcommand of the email command, and configures email.

Synopsis

```
$ cscli service_console configure email add -h
```

Usage:

```
cscli service_console [options] configure [options] email [options] add [options]
```

Optional Arguments	Description
-A |--address	E-mail address (required)
-h |--help	Displays the help message and exits

configure email_delete Subcommand

The `delete` command is a second-level subcommand of the `email` command, and is used to configure email.

Synopsis

```
$ cscli service_console configure email delete -h
```

Usage:

```
cscli service_console [options] configure [options]
email [options] delete [options]
```

Optional Arguments	Description
-A |--address	E-mail address (required)
-h |--help	Displays the help message and exits

Additional context:

```
$ cscli service_console configure email show
No email addresses are configured to receive service notifications.
$ cscli service_console configure email add -A john.smith@seagate.com
Successfully added 'john.smith@seagate.com' to receive notification emails.
$ cscli service_console configure email add -A bob.smith@seagate.com
Successfully added 'bob.smith@seagate.com' to receive notification emails.
$ cscli service_console configure email add -A junk@example.com
Successfully added 'junk@example.com' to receive notification emails.
$ cscli service_console configure email add -A junk@example.com
Unable to add: 'junk@example.com' is already configured for receiving notification
emails.
$ cscli service_console configure email delete -A junk@example.com
Successfully removed 'junk@example.com' from receiving notification emails.
$ cscli service_console configure email delete -A junk@example.com
Unable to delete: 'junk@example.com' is not configured for receiving notification
emails.
$ cscli service_console configure email show
The following email addresses are configured to receive service notifications:
john.smith@xyzcorp.com
bob.smith@xyzcorp.com
$ cscli service_console configure email disable
```

ok.

```
$ cscli service_console configure email enable
```

Or an error example:

```
$ cscli service_console configure email send_test_email
Failed to send test e-mail to configured user(s): an error has occurred.
```

17.25 System Software Update Commands

The following commands prepare a software upgrade package for installation and apply it to system nodes.

`update_node` Command

Modes: Daily

The `update_node` command updates software on the specified node(s).

Synopsis

```
$ cscli update_node [-h] -n node_spec
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-n` *node_spec*`|--node-spec` *node_spec*	Specifies hostnames of the nodes on which to update software.

`set_node_version` Command

Modes: Daily

The `set_node_version` command is used to change the image of diskless node(s). It is used to boot a given set of diskless appliance nodes into the specified version of Sonexion appliance for a system upgrade. This command is available for the "admin" account only.

Synopsis

```
$ cscli set_node_version [-h] -n nodes -v version
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-n` *nodes* `|--node` *nodes*	Displays pdsh-style nodes hostnames.
`-v` *version* `|--version` *version*	Version of the image to use.

17.26 `support_bundle` Command

The `support_bundle` command manages support bundles and support bundle settings.

When a support bundle is collected, it contains extra information about RAID configuration, and local Lustre users (if any are defined). Additional log files are available in the support bundle for MDRAID examine output and lustre users/groups.

Synopsis

```
$ cscli support_bundle [-h] {collect,set,export,show,delete}
```

Positional Arguments	Description
`collect`	Collects the support bundle for specified nodes.
`show`	Support bundle `show` command.
`export`	Export support bundles as the **Tar.GZ** archive (into the current folder).
`set`	Support bundle `set` command.
`delete`	*(Added for release 3.0.)* Delete support bundle.

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.

support_bundle collect Subcommand

The `support_bundle collect` subcommand displays help, pdsh-style node names, and the time window in minutes.

Synopsis

```
$ cscli support_bundle collect [-h] [-n nodes] [-t minutes]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-n` *`nodes`*`|--nodes` *`nodes`*	Displays pdsh-style node names. The default is all nodes.
`-t` *`minutes`*`|--time-window` *`minutes`*	Displays the time window in minutes. The default is 45 minutes.

support_bundle show Subcommand

The `support_bundle show` subcommand displays help, triggers that initiate automatic bundle collection, a list of support bundles collected, and the purge limit.

Synopsis

```
$ cscli support_bundle show [-h] (--triggers | --bundles | --purge-limit)
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-t |--triggers`	Displays triggers that initiate automatic bundle collection.

Optional Arguments	Description
`-b |--bundles`	Displays a list of support bundles collected.
`-p |--purge-limit`	Displays the purge limit. Free file system space limit in percents, after reaching which Trinity will purge old support bundle files.

support_bundle export Subcommand

The `support_bundle export` subcommand displays help, and the bundle ID.

Synopsis

```
$ cscli support_bundle export [-h] bundle_id
```

Optional Arguments	Description
bundle_id	ID number of the support bundle file, which can be obtained using the `cscli support_bundle show` command.
`-h |--help`	Displays the help message and exits.

support_bundle set Subcommand

The `support_bundle set` subcommand displays help, purge limit for the support bundle, and triggers that initiate automatic bundle collection.

Synopsis

```
$ cscli support_bundle set [-h]
(-p purge_limit | --trigger {lbug,Failover})
[--on | --enable | --off | --disable]
```

Optional Arguments	Description
`-h |--help`	Displays the help message and exits.
`-p` *purge_limit* `|--purge-limit` *purge_limit*	Sets the purge-limit in percentage for support bundles.

`-t |--trigger {`*lbug*`,Failover},`
`{`*lbug*`,Failover}`
	Displays triggers that initiate automatic bundle collection.

`--on`	Turns the command on.
`--enable`	Enables the command.
`--off`	Turns the command off.
`--disable`	Disables the command.

18 GEM CLI Commands

This section describes command line commands provided by the Generic Enclosure Management (GEM) software. While these commands are not typically used during normal operation, they can be used to help resolve issues.

The GEM software controls and monitors the hardware infrastructure and overall system environmental conditions. GEM manages system health, provides power control to power-cycle major subsystems, monitors fans, thermals, power consumption, etc., and offers extensive event capture and logging mechanisms to support file system failover and enable post-failure analysis of hardware components.

Each enclosure and FRU within has an EEPROM containing Vital Product Data (VPD) that is used by GEM to identify and control system components. GEM can pars three main VPD types:

- Midplane
- Power/Cooling Module (PCM)
- Canister

Unified Sytem Management (USM) combines the GEM firmware with firmware from other enclosure components, such as x86 subsystems, primarily BIOS, BMC, and FPGA.

Serial port settings

Use the following settings for using HyperTerminal or other serial communications GUI to work with the CLI: Baud rate (bits/sec):

Baud rate (bits per second):	115200
Data bits:	8
Parity:	None
Stop bits:	1
Flow control:	None

The above settings apply to manually typed commands. If multiple commands are sent via a text file, then the baud rate needs to be reduced for all characters to be processed.

Set the baud rate in the running firmware by issuing:

```
rmon baud 0
```

Change the serial communications GUI settings to: Baud rate (bits/sec):

Baud rate (bits/sec):	9600
Data bits:	8

Parity:	None
Stop bits:	1
Flow control:	None

To return to the higher baud rate, issue:

```
rmon baud 4
```

The complete set of supported values is:

0 = 9600

1 = 19200

2 = 38400

3 = 57600

4 = 115200

Supported number bases

Numeric parameters passed into CLIs can be in different bases. Decimal is the default. Octal or hexadecimal can be supplied by using a leading code:

Decimal – Plain number

Octal – Leading ‘0’

Hexadecimal – Leading ‘0x’

For example, the decimal number 14 would be represented in the following ways:

Decimal – 14

Octal – 016

Hexadecimal – 0xE

18.1 ddump

Command name:	`ddump`
Command synopsis:	Returns a system-wide diagnostic dump
Command description:	Calls all commands of the command type 'diagnostic' that do not demand an argument; that is, a simple single-shot diagnostic dump.
Command arguments:	None
Command type:	Diagnostic
Access level:	General

18.2 getboardid

Command name:	`getboardid`
Command synopsis:	Reports the local board slot ID and HA mode
Command description:	Reports the local board slot ID and HA mode in human-readable and machine-readable form.
Command arguments:	hex: Returns the slot ID (byte 1) and HA mode (byte 2) in hexadecimal form. If the canister is the master, then the HA mode is set to 0x0. If the canister is the slave, then the mode is 0x00.
Command type:	Debug
Access level:	General

18.3 getmetisstatus

Command name:	`getmetisstatus`
Command synopsis:	Reports Metis status for the enclosure. (Supplies reserve power to protect in-flight storage data, enabling it to be securely stored on persistent media).
Command description:	Invoking this command returns Metis status in human-readable or machine-readable form.
Command arguments:	Argument 1 [hex]: If the "hex" argument is present, the Metis status is reported in machine-readable form. If "hex" is not specified, the status is reported in human-readable form.
Command type:	Diagnostic
Access level:	Engineering

18.4 getvpd

Command name:	`getvpd`
Command synopsis:	Retrieves VPD information from all enclosure FRUs
Command description:	The `getvpd` command displays the following enclosure VPD data: ● Enclosure Vendor

	● Enclosure Product ID ● Enclosure WWN ● Enclosure Serial Number ● Enclosure Part Number ● Canister VPD Version ● Canister Vendor ● Canister Product ID ● Canister SAS Address ● Canister Serial Number ● Canister Part Number ● Midplane VPD Version ● Midplane Product ID ● Midplane Serial Number ● Midplane Part Number ● PCM VPD Version ● PCM Vendor ● PCM Product ID ● PCM Serial Number ● PCM Part Number
Command arguments:	`getvpd` – No additional arguments
Command type:	Debug
Access level:	General

18.5 help

Command name:	`help`
Command synopsis:	Displays helpful information about the GEM commands
Command description:	Provides a mechanism to discover the available commands and display the command usage information. By default (i.e. no argument supplied), the command only lists the synopsis for those commands with the access level 'general'. The argument all lists the synopsis for all commands, regardless of access level. The argument testing lists the synopsis for all commands that have the 'testing' access level. If the argument matches a command (for example help ddump) then detailed help for the specified command displays instead.
Command arguments:	One optional argument - see description above.

Command type:	Control
Access level:	General

18.6 ipmi_power

Command name:	`ipmi_power`
Command synopsis:	Performs safe canister-level power control using chassis commands to the BMC
Command description:	This command allows the user to request a canister-level shutdown through the BMC. The benefit of using this command is to cleanly shut down the x86 subsystem using ACPI.
Command arguments:	2 | "soft" – Orchestrated shutdown of x86 complex. 3 | "off" – Immediate shutdown of x86 complex. 4 | "cycle" – Canister power cycle. 5 | "reset" – Canister reset. 6 | "on" – Wake x86 complex from standby/soft-off.
Command type:	Control
Access level:	General Access

18.7 ipmi_setosboot

Command name:	`ipmi_setosboot`
Command synopsis:	Sets a value in the IPMI OS boot sensor indicating that the x86 subsystem has successfully booted. The OS boot sensor value is cleared to zero (0) on x86 resets and BMC firmware upgrades / reboots.
Command description:	This command is intended for use by an application on the local x86 subsystem to set the OS boot sensor to confirm that the system has finished booting and the OS is in full control. This command MUST be invoked by the customer OS on startup. If it is not set and GEM detects an AC loss event, then the module is automatically shut down. This shutdown ensures that the system batteries are not flattened by a module booting at full power. Without a parameter, the command reads the current sensor value. With a parameter of 1, the command sets the sensor to indicate that the system has booted (0x40) and then reads back the sensor for confirmation.
Command arguments:	`ipmi_setosboot` [*`setting`*]

Command type:	Control
Access level:	Engineering

18.8 logdump

Command name:	`logdump`
Command synopsis:	Displays logged messages
Command description:	Provides a mechanism to output logging information.
Command arguments:	6 optional arguments: Argument 1 specifies the area of memory from which to retrieve log messages from. 'r' = RAM, 'n' = non-volatile. Argument 2 specifies the order of the log messages. "old" = oldest first, "new" = newest first. Argument 3 limits the number of logged messages displayed to *`n`*. Set to zero (0) or omit the argument to display all logged messages. Argument 4 controls the generation of a *`timestamp`* field in the log dump messages. Set to 1 for enable; 0 for disable. Argument 5 controls the generation of a *`subsystem name`* field in the log dump messages. Set to 1 for enable; 0 for disable. Argument 6 controls the generation of a *`service name`* field in the log dump messages. Set to 1 for enable; 0 for disable. The default (for omitted command arguments) displays all logged messages from RAM, newest first, with all message fields enabled.
Command type:	Diagnostic
Access level:	General

18.9 report_faults

Command name:	`report_faults`
Command synopsis:	Reports all system-wide faults
Command description:	Outputs all known faults, collected from each GEM service.
Command arguments:	None
Command type:	Diagnostic

Access level:	General

18.10 settime Command

Command name:	`settime`
Command synopsis:	Sets GEM logging time in days, hours, minutes and seconds
Command description:	`settime` *`days hh mm ss`* For example: `settime 10 9 8 7` sets the logging time to 10 days, 9 hours, 8 minutes and 7 seconds. The new logging time appears in the log timestamps as: 10+09:08:07.123 M0 > Using the `settime` command on its own, without any arguments, prints the current logging time to the CLI.
Command arguments:	*`days hh mm ss`*
Command type:	Control
Access level:	General

18.11 ver

Command name:	`ver`
Command synopsis:	Displays version information
Command description:	Displays version numbers and information for the components in the local canister, midplane and PCMs.
Command arguments:	None
Command type:	Diagnostic
Access level:	General