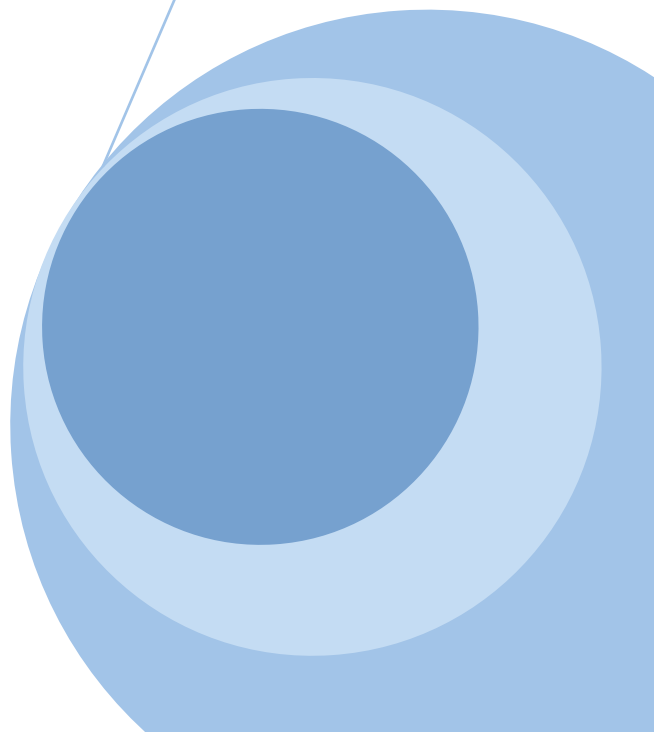


Sonexion[®] Administrator's Guide

Software Version 1.5.0

S-2537-15



© 2014 Cray Inc. All Rights Reserved. This manual or parts thereof may not be reproduced in any form unless permitted by contract or by written permission of Cray Inc.

U.S. GOVERNMENT RESTRICTED RIGHTS NOTICE

The Computer Software is delivered as “Commercial Computer Software” as defined in DFARS 48 CFR 252.227-7014. All Computer Software and Computer Software Documentation acquired by or for the U.S. Government is provided with Restricted Rights. Use, duplication or disclosure by the U.S. Government is subject to the restrictions described in FAR 48 CFR 52.227-14 or DFARS 48 CFR 252.227-7014, as applicable. Technical Data acquired by or for the U.S. Government, if any, is provided with Limited Rights. Use, duplication or disclosure by the U.S. Government is subject to the restrictions described in FAR 48 CFR 52.227-14 or DFARS 48 CFR 252.227-7013, as applicable.

TRADEMARKS

Cray and Sonexion are federally registered trademarks and Active Manager, Cascade, Cray; Apprentice2, Cray; Apprentice2; Desktop, Cray; C++; Compiling; System, Cray; CX, Cray; CX1, Cray; CX1-iWS, Cray; CX1-LC, Cray; CX1000, Cray; CX1000-C, Cray; CX1000-G, Cray; CX1000-S, Cray; CX1000-SC, Cray; CX1000-SM, Cray; CX1000-HN, Cray; Fortran; Compiler, Cray; Linux; Environment, Cray; SHMEM, Cray; X1, Cray; X1E, Cray; X2, Cray; XD1, Cray; XE, Cray; XEm, Cray; XE5, Cray; XE5m, Cray; XE6, Cray; XE6m, Cray; XK6, Cray; XK6m, Cray; XMT, Cray; XR1, Cray; XT, Cray; XTm, Cray; XT3, Cray; XT4, Cray; XT5, Cray; XT5, Cray; XT5m, Cray; XT6, Cray; XT6m, CrayDoc, CrayPort, CRInform, ECOPhlex, LibSci, NodeKARE, RapidArray, The Way to Better Science, Threadstorm, uRiKA, UNICOS/lc, and YarcData are trademarks of Cray Inc.

AMD and Opteron are registered trademarks of Advanced Micro Devices, Inc. Xilinx is a registered trademark of Xilinx, Inc. Linux is a registered trademark of Linus Torvalds. All other trademarks are the property of their respective owners.

All other trademarks are the property of their respective owners.

Direct requests for copies of publications to:

Mail: Cray Inc. Logistics
PO Box 6000
Chippewa Falls, WI 54729-0080
USA

E-mail: spares@cray.com

Fax: +1 715 726 4602

Direct comments about this publication to:

Mail: Cray Inc.
Technical Training and Documentation
P.O. Box 6000
Chippewa Falls, WI 54729-0080
USA

E-mail: ttd_online@cray.com

Fax: +1 715 726 49

Record of Revision

Publication Number	Description
HR5-6093-0	August 2012: Original Printing
HR5-6093-A	September 2012 : Release 1.2
HR5-6093-B	April 2013: Release 1.2.1
S-2537-131	March 2014: Release 1.3.1
S-2537-131a	April 2014: Most of section 3 removed. LNET configuration revised.
S-2537-14	May 2014: Release 1.5.0. Help system content removed. Changes to Support Bundles , CLI.
S-2537-14a	July 2014: Revised section 4. Old section 7 removed.
S-2537-15	October 2013: Release 1.5.0

Contents

1. Introducing Sonexion	7
1.1 Software Architecture	7
1.2 Hardware Architecture	9
2. What's New?	12
2.1 New software features	12
2.2 New hardware features	13
3. Custom LNET configuration	14
4. Change Network Settings	17
4.1 Change the DNS Resolver Configuration	17
4.2 Change Externally Facing IP Addresses	18
4.3 Change the LDAP settings in Daily Mode	19
4.4 Configure NIS Support in Daily Mode	21
4.5 Change NIS Settings in Daily Mode	22
5. Support Bundles	24
5.1 Support file overview	24
5.2 Manually collect support files	26
5.3 Viewing support files	29
5.4 Use CSCLI for support bundles	34

5.5	Interpret Sonexion support bundles	35
6.	Troubleshooting.....	38
6.1	CSSM Issues.....	38
6.2	Networking Issues	41
6.3	RAID/HA Issues.....	42
6.4	Other Issues	52
7.	CSCLI User Documentation	55
7.1	CSCLI Overview	55
7.2	Summary of Changes in Release 1.5.0	56
7.3	CLI command summary.....	58
7.4	Network Setup Commands.....	61
7.5	Lustre User Commands.....	63
7.6	System Alert Commands.....	76
7.7	Node Control Commands	88
7.8	Administrative Commands.....	92
7.9	Configuration Commands	97
7.10	Service Commands	99
7.11	Filter Commands	100
7.12	Update System Software	101
7.13	Ibstat Check Command	102
7.14	Manage Nodes	103
7.15	Manage Node Position in a Rack.....	104
7.16	Monitor the System's Health	105
7.17	RAID Management	115
7.18	Administrative Settings	117
7.19	Managing Support Bundles	120
8.	GEM CLI Commands.....	123
8.1	Serial port settings.....	123
8.2	Supported number bases	124
8.3	Supported commands	124

Tables

Table 1. LDAP Settings	19
Table 2. CLI Changes for Release 1.5.0	56
Table 3. CLI Command Summary	58
Table 4. New and Old luster_users Commands Comparison.....	75

1. Introducing Sonexion

This Administration Guide provides step-by-step instructions on how to set up, use, and troubleshoot the Cray Sonexion storage system. This manual is intended for Site Service Providers who maintain Cray Sonexion storage systems.

1.1 Software Architecture

Sonexion software architecture consists of an integrated, multi-layer software stack:

- Cray Sonexion System Manager (CSSM)
- Lustre file system
- Data protection layer (Redundant Array of Independent Disks, RAID)
- Unified System Management (USM)
- Linux OS

Sonexion runs Lustre 2.1.0 software in a standard Linux environment (Scientific Linux 6.2 operating system). The file system is fully integrated with CSSM, USM, and RAID layers in the stack.



1.1.1 Cray Sonexion System Manager (CSSM)

Cray Sonexion System Manager (CSSM) provides a single-pane-of-glass view of the Sonexion solution infrastructure. It includes a browser-based GUI that simplifies cluster installation and configuration, and provides consolidated management and control of the entire storage cluster. CSSM also provides distributed component services to manage and monitor system hardware and software.

CSSM includes wizards to guide you through configuration and node provisioning. Once the cluster is running, you can use the GUI to manage the storage environment with these functions:

- Start and stop file systems
- Manage node failover
- Monitor node status
- Collect and browse performance data

The dashboard reports errors and warnings for the storage cluster and provides tools to aid in troubleshooting, including cluster-wide statistics, system snapshots, and Lustre syslog data.

To maximize availability, CSSM works with USM, the platform's integrated management software, to provide comprehensive system health monitoring, error logging, and fault diagnosis. Users are alerted to changing system conditions and degraded or failed components.

1.1.2 Data Protection Layer (RAID)

Sonexion uses Redundant Array of Independent Disks (RAID) to provide different data protection layers throughout the system. For new builds, the RAID subsystem has been changed to GridRAID technology. See "New software features", page 12.

1.1.3 Unified System Management (USM) firmware

Extensive Sonexion system diagnostics are managed by USM management firmware, which runs on each OSS in the scaleable storage unit (SSU). USM monitors and controls the SSU's hardware infrastructure and overall system environmental conditions, providing a range of services including SES and high-availability (HA) capabilities for system hardware and software. USM offers these key features:

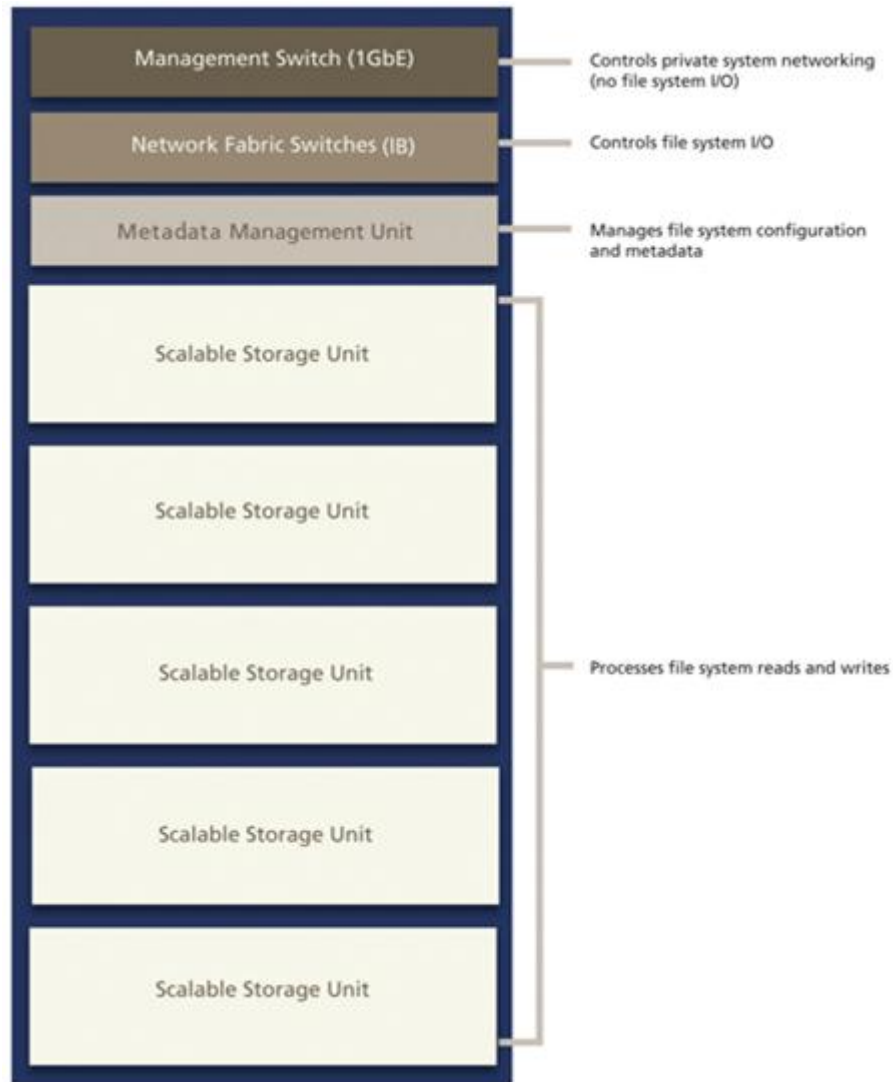
- Manages system health, providing random-access services (RAS) that cover all major components such as disks, fans, power-supply units (PSUs), Serial Attached SCSI (SAS) fabrics, PCI (Peripheral Component Interconnect) buses, memories, and CPUs, and provides alerts, logging, diagnostics, and recovery mechanisms
- Power control of hardware subsystems that can be used to individually power-cycle major subsystems and provide additional RAS capabilities – this includes drives, servers, and enclosures
- Fault-tolerant firmware upgrade management
- Monitoring of fans, thermals, power consumption, voltage levels, AC inputs, field-replaceable unit (FRU) presence, and health
- Efficient adaptive cooling keeps the SSU in optimal thermal condition, using as little energy as possible
- Extensive event capture and logging mechanisms to support file system failover capabilities and to allow for post-failure analysis of all major hardware components

1.2 Hardware Architecture

The Sonexion 1600 hardware architecture consists of a pre-configured, rack-level storage cluster that can be easily expanded using modular storage node building blocks. The principal hardware components include:

- Metadata Management Unit (MMU)
- Scalable Storage Unit (SSU)
- Network Fabric Switches
- Management Switch

The Sonexion solution is differentiated from other file system solutions by its innovative MMU and SSU architectures.



1.2.1 Metadata Management Unit

The Metadata Management Unit (MMU) is a quad-node server which contains the two management (MGMT) nodes, the MGS node, the MDS node, and one shelf of high-availability shared storage. The central point of management for the entire storage cluster, the MMU runs CSSM software, manages network request handling, and monitors every storage element within the cluster. Sonexion interface ports support InfiniBand fabric network interface technology connections and 1GbE management network connections.

The MMU is fully redundant and fault-tolerant. Each node is configured for active-passive failover, with an active instance of the server running on one system and a passive instance of the node running on the peer system. If an active node fails, for example, the MDS goes down, then the passive MDS node takes over the MDT operations of the failed MDS. The shared storage of the MMU supports a combination of Small Form Factor (SFF) SAS

HDD, protected using RAID 1, for management data, file system data, and journal acceleration.

Sonexion supports InfiniBand connections to the MGMT, MDS, and MGS nodes. Additionally, each server connects, via Ethernet, to dedicated private management networks supporting Intelligent Platform Management Interface (IPMI).

1.2.2 Scalable Storage Unit

The core building block is the Scalable Storage Unit (SSU). Each SSU is configured with identical hardware and software components, and hosts two OSS nodes. The SSU contains two OSSs, with RAID-protected, high availability shared storage, and interface ports to support InfiniBand data networks and 1GbE management network connections.

The OSSs are Storage Bridge Bay (SBB) compliant with SAS expanders that enable both modules to directly access all drives in the enclosure (a differentiator among fully fault-tolerant systems). The OSSs connect through a common midplane, eliminating the need for extra external cables, and share a redundant, high-speed interconnect across the midplane for failover services.

The SSU is fully redundant and fault-tolerant, thus ensuring maximum data availability. Each OSS serves as a Lustre node, accessing the disk as shared OST storage and providing active-active failover. If one OSS fails, the active module manages the OSTs and the disk operations of the failed node. In non-failure mode, the I/O load is balanced between modules. The SSU's shared storage consists of high-capacity SAS disk drives, configured in a RAID 6 array to protect against double disk failures and drive failure during rebuilds.

1.2.3 Network fabric switches

The Network Fabric switches (InfiniBand) manage I/O traffic and provide network redundancy throughout the Sonexion solution. To maximize network reliability, the OSSs in the SSU are connected to redundant network switches. If one switch fails, the second module in the SSU (connected to the active switch) manages the OSTs of the module connected to the failed switch.

To maintain continuous management connectivity within the solution, the network switches are fully redundant at every point and interconnected to provide local access from the MGMT, MDS, and MGS nodes to all storage nodes.

1.2.4 Management switches

The Management switches consists of a dedicated local network on dual 1GbE switches that is used for configuration management and health monitoring of all components in the Sonexion solution. The management network is private and not used for data I/O in the cluster. This network is also used for IPMI traffic to the SSU's OSSs, enabling them to be power-cycled by CSSM.

2. What's New?

The following is a list of new features and improvements for this release of CSSM version 1.5.0.

2.1 New software features

- **GridRAID**

GridRAID, the Sonexion implementation of Parity Declassed RAID (PDRAID), is a RAID organization that combines the logical structure of 8+2 RAID 6 data protection with the pseudorandom distribution of these RAID 6 parity groups, along with reserved spare space across a large number of physical storage devices.

This allows accelerated restoration of redundancy, referred to as “GridRAID reconstruct” (phase 1), and a modified “GridRAID rebalance” (phase 2) that restores data for a failed drive onto a physical replacement drive. A comparison:

- MDRAID (previous method): each OSS typically operates using 40 drives configured as four 8+2 RAID 6 arrays (4 OSTs per OSS and 8 OSTs total per SSU), plus 2 global hot spares for a total of 82 drives.
- GridRAID (new method): 82 drives are configured as two 41-drive GridRAID arrays with no dedicated hot spare drives. Two drives’ worth of spare space is distributed across all 41 drives in the array.

- **New and improved monitoring dashboard**

The new Dashboard replaces the dashboard functionality from previous releases and provides a high level view into the entire storage cluster providing top level knowledge into the overall behavior and health of the system. The dashboard displays the

following widgets that report high-level system health and issues with individual nodes to aid in troubleshooting and resolving problems quickly:

- Node Status
- File System Throughput
- Inventory
- Top System Statistics
- **SSU+n systems containing one SSU enclosure and up to three ESU enclosures**
 - The SSU+n feature, where the maximum value for n=3, whereby up to 3 Expansion Storage Units (ESUs) can be added to each SSU, for this release provides additional capacity at reduced cost.
 - This feature is available only for newly installed systems. The ability to add ESUs to existing systems will be available in a future release.
- **GUI guest account**

A built-in guest account for read-only access to CSSM is available in this release.
- **NIS GUI support**

Added GUI support for configuring NIS as an option for Lustre users.
- **USM 2.16 firmware update**
- **Lustre features**
 - 4MB IO - Support for larger than 1MB sequential I/O RPCs. This requires a Lustre 2.4 Client.
 - Extended Attribute Cache for Lustre (XATTR Cache) - The patch implements an extended attribute cache for a Lustre client. It is organized as a write-through cache: reads are performed from cache, updates are sent synchronously to the MDS. An additional inode bit is added to protect the cache.

2.2 New hardware features

- The SSU+n configuration where the maximum value of n is 3, refers to an SSU with 3 ESUs attached. The SSU+n configuration consists of the following:
 - One Scalable Storage Unit (SSU) and up to three ESU (Expansion Storage Unit)
 - Maximum of 32 OSTs configured on the SSU for MDRAID at 8 OSTs per enclosure (8 OSTs x 4 enclosures = 32 OST's managed by the SSU)
 - Maximum of 8 OSTs configured on the SSU for GridRAID at 2 OSTs per enclosure (2 OSTs x 4 enclosures =8 OST's managed by the SSU)
- Each SSU and its companion ESU must be located vertically adjacent in the same rack

3. Custom LNET configuration

Use the procedure in this chapter to configure a custom LNET configuration on the Sonexion system while in “daily mode” (see section 0, page 55).

For a new system, first follow the setup procedures described in Cray publication HR5-6124, *Sonexion Field Installation Guide*. Then execute the following installation.

1. Log in to the primary management mode.

2. Change to root:

```
$ sudo su -
```

3. Stop the Lustre file system by running the command:

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

4. For version 1.2:

- a. Log in to the MGS node via SSH.
- b. If you do not know the MGS group, run this command:

```
crm_mon -l -r
```

The group with **md65** in its name is the MGS group.

- c. Stop the MGS service, by entering:

```
# stop_xyraid mgs_group
```

- d. Log out of the MGS node and back into primary MGMT server node.

5. To change the **o2ib** index, use the following steps:

a. Start the MySQL client and connect to the **t0db** database, by entering:

```
# mysql t0db
```

b. Display the **mgsNID**, **nidFormat**, and **nidIndex** entries by entering:

```
mysql> select * from property where name in ('nidFormat',
      'mgsNID', 'nidIndex');
```

Following is a sample output for versions 1.3.1 and 1.5.0:

```
mysql> select * from property where name in ('nidFormat', 'mgsNID',
      'nidIndex');
```

id	context	name	value	attr_type
22	snx11033n:beConfig	nidFormat	l%s@o2ib%d	str
106	snx11033n:beConfig	nidIndex	3	int
109	snx11033n:beConfig	mgsNID	lsnx11033n002@o2ib3	str

```
3 rows in set (0.00 sec)
```

c. Set the **o2ib** index by modifying the **nidIndex** entry by specifying:

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

Sample:

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

d. Set the **mgsNID** entry to match the **o2ib** index by entering:

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

Sample output:

```
mysql> update property set value='lsnx11033n002@o2ib2' where
      name='mgsNID';
Query OK, 1 row affected (0.04 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

- e. Verify the changes by repeating step 5b.

Sample output:

```
mysql> select * from property where name in ('nidFormat', 'mgsNID',
      'nidIndex');
```

id	context	name	value	attr_type
22	snx11033n:beConfig	nidFormat	l%so2ib%d	str
106	snx11033n:beConfig	nidIndex	2	int
109	snx11033n:beConfig	mgsNID	lsnx11033n002@o2ib2	str

3 rows in set (0.00 sec)

- f. Close the MySQL session by specifying:

```
mysql> quit
```

- g. Run puppet:

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

6. Run the 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
```

7. Verify that the customized LNET configuration has been applied.

- a. List the node NIDs by entering:

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

- b. List the nodes and targets, by entering:

```
# cscli fs_info
```

8. For version 1.2 **only**, run:

```
# pdsh -g lustre mdraid-deactivate
# pdsh -g lustre manage_all_xyraid
```

9. Start the Lustre file system by entering:

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

Wait for the targets to mount on all system nodes.

This completes the procedure.

4. Change Network Settings

This chapter contains procedures for changing several different network settings on Sonexion. The following topics are included:

- Changing the DNS resolver configuration
- Changing the externally facing IP addresses
- Changing the LDAP settings in Daily Mode
- Configuring NIS Support in Daily Mode
- Changing NIS Settings in Daily Mode

4.1 Change the DNS Resolver Configuration

This procedure is used to change the DNS resolver, the system service which translates URLs into IP addresses.

1. SSH into the primary MGMT node. Run:

```
$ ssh -l admin primary_MGMT_node
```

2. Sudo to root:

```
[MGMT0]$ sudo su -
```

3. Update the DNS settings in the **t0db** database. Run:

```
[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. Run:

```
[MGMT0]# /opt/xyratex/bin/beUpdatePuppet -s -g mgmt
```

4.2 Change Externally Facing IP Addresses

Some customers may wish to change one or both of the externally facing 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.

To change this configuration, follow the steps given below:

1. Log in to the secondary MGMT node
2. SSH to the primary MGMT node by entering:

```
$ ssh -l admin primary_MGMT_node
```

3. Sudo to root:

```
[admin@n000]$ sudo su -
```

4. Edit the Ethernet configuration file. Run:

```
[admin@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 it 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 valid IP address). Change this file as desired.

5. Toggle the Ethernet interface. Run:

```
[root@n000]# ifdown eth1; ifup eth1
```

6. Exit out of the SSH session for primary MGMT node
7. Log in to the primary MGMT node and SSH into the secondary MGMT node.
8. Repeat steps 3-5 to configure the externally facing IP address on the secondary MGMT node.

4.3 Change the LDAP settings in Daily Mode

LDAP settings 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 1. 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_dn	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.

1. SSH into the primary MGMT node. Run:

```
$ ssh -l admin primary_MGMT_node
```

2. Sudo to root:

```
[MGMT0]$ sudo su -
```

3. Print the existing configuration as follows:

- a. Enter:

```
[MGMT0]# cat /etc/nslcd.conf
```

Sample output:

```
[root@snx11000n000 ~]# 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. Enter:

```
[MGMT0]# mysql t0db -e "select * from ldap_setup;"
```

Sample output:

```
[root@snx11000n000 ~]# mysql t0db -e "select * from ldap_setup;"
```

ldap_setup_id	server_name	port	base_dn	user_dns	group_dns	hosts_dns	bind_dn	password	cluster_id
1	172.30.12.19	389	dc=datacenter, dc=cray,dc=com	ou=People, dc=datacenter, dc=cray,dc=com			NULL	NULL	1

4. Change the base DN:

```
[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).

4.3.1 LDAP over TLS Configuration Settings in Daily Mode

Set up an LDAP/TLS server. Perform the following to get LDAP over TLS working on release 1.3.1:

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 your 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 your 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/1.3.1-18/appliance.x86_64/etc/openldap/cacerts/
```

4.4 Configure NIS Support in Daily Mode

This section provides instructions for configuring the Network Information Services (NIS) client for Sonexion. This procedure applies to releases in 1.3.1 and 1.5.0. Sonexion 1.3.1 supports LDAP and NIS but was only intended to support configuration of NIS during the Customer Wizard phase of the installation.

4.4.1 Requirements

NIS must not have been previously been configured during Site Configuration mode, otherwise see section 4.5, “Change NIS Settings in Daily Mode”, page 22.

You will need to know the NIS Domain name, and the IP address of NIS servers reachable (pingable) from the Management Nodes.

4.4.2 NIS Installation

1. Enable support for NIS:

```
[root@n000]# mysql t0db -e 'update filesystem set lustre_upcall="nis"'
```

2. Configure in 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")'
```

If you wish 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", "xxxxxxxx", "str")'
```

Where `xxxxxxxx` is the value of `nisdomainname` on the relevant server.

4. The above changes must be followed by:

```
[root@n000]# beUpdatePuppet -sa
```

5. Once this has been done, check that `/etc/puppet/data/CSSM/nis.yaml` has been updated on 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: xxxxxxxx
nis_server: [xx.xx.xx.xx]
```

6. Check `/etc/yp.conf` on MMU nodes contain 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 XXXXXXXXXXXX server xx.xx.xx.xx
```

7. Check whether `ypbind` is running on all MMU nodes
8. Final checks to be run on all MMU nodes:

```
[root@n000]# service ypbind status
[root@n000]# ypwhich
[root@n000]# ypwhich -m
```

4.5 Change NIS Settings in Daily Mode

This section provides instructions to change the NIS settings in daily mode, assuming that NIS has already been configured. This procedure applies to releases 1.3.1 and 1.5.0.

4.5.1 Requirements

NIS has previously been configured, either during customer Wizard or following instructions above in section 4.4, “Configure NIS Support in Daily Mode”, page 21.

4.5.2 NIS Configuration

1. SSH into the primary MGMT node. Run:

```
$ 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]# myql t0db -e 'update property set value = "xx.xx.xx.xx
yy.yy.yy.yy" where name = "nis_server"'
```

You can specify more one or more IP addresses for NIS master and its reachable NIS slaves.

5. Change the NIS domain name of the NIS server:

```
[root@n000]# myql t0db -e 'update property set value = "xxxxxxxx"
where name = "nis_domain"'
```

6. The above changes must be followed by:

```
[root@n000]# beUpdatePuppet -sa
```

7. Verify the new NIS server settings by repeating step 3.

5. Support Bundles

This section discusses the creation and use of *support bundles*, which are collections of event logs from field systems. Support bundles are used to debug many Sonexion problems and are collected by Cray personnel. Sonexion provides a mechanism for collecting support bundles, which may be initiated manually or triggered automatically by certain events (for example, Lustre bugs and failover events). These support bundles should be provided to Cray personnel in the course of requesting technical support.

When you encounter a problem, refer to the following topics to assist with resolving the issue. Use the following tabs in the CSSM GUI:

- 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.

5.1 Support file overview

The CSSM **Support** tab provides support functionality that lets you collect diagnostic information, including logs and configuration settings, on an automatic or manual basis. When a Lustre error occurs, the system automatically collects diagnostic information. Alternately, 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.

This section describes the data collection process and contains procedures to work with the diagnostic information.

5.1.1 Collecting Sonexion data in support files

When a Lustre error or a system event (such as failover) occurs, Sonexion automatically triggers a process to collect a 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 collecting 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 will not start a second data collection process related to the later error.

5.1.2 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

5.1.3 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.

You may also manually start a support bundle collections 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.

5.2 Manually collect support files

The Sonexion user can manually start system data collection and create support files. Unlike the automatic process, the manual data collection operation can be canceled.

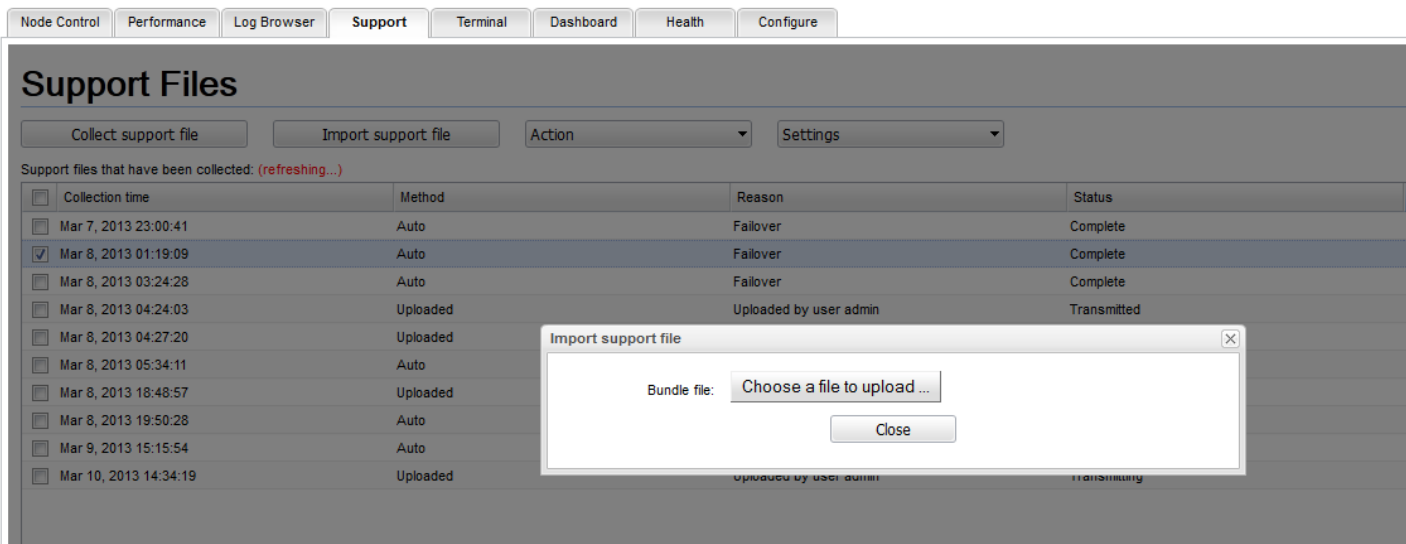
To manually collect Sonexion support files:

1. In CSSM, click the **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.



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 will select 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 and the support file is created, which is a zip file containing hundreds of different log files.

5.2.1 Import a support file

Use the Import feature to upload a single support file bundle into CSSM to view its contents.

To import a support file:

1. In CSSM, click the **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.

The screenshot shows the 'Support Files' section of the CSSM interface. At the top, there are navigation tabs: Node Control, Performance, Log Browser, Support (selected), Terminal, Dashboard, Health, and Configure. Below the tabs, there are buttons for 'Collect support file', 'Import support file', and 'Action', along with a 'Settings' dropdown menu. A table lists support files that have been collected, with columns for 'Collection time', 'Method', 'Reason', and 'Status'. The table contains several rows, with the second row (Mar 8, 2013 01:19:09) selected. An 'Import support file' dialog box is open in the foreground, showing a 'Bundle file:' label and a 'Choose a file to upload ...' button, with a 'Close' button below it.

Collection time	Method	Reason	Status
Mar 7, 2013 23:00:41	Auto	Failover	Complete
Mar 8, 2013 01:19:09	Auto	Failover	Complete
Mar 8, 2013 03:24:28	Auto	Failover	Complete
Mar 8, 2013 04:24:03	Uploaded	Uploaded by user admin	Transmitted
Mar 8, 2013 04:27:20	Uploaded		
Mar 8, 2013 05:34:11	Auto		
Mar 8, 2013 18:48:57	Uploaded		
Mar 8, 2013 19:50:28	Auto		
Mar 9, 2013 15:15:54	Auto		
Mar 10, 2013 14:34:19	Uploaded		

5.2.2 Download a support file

Use the download a copy feature to save a local copy of the selected support file.

To download a support file:

1. In CSSM, click the **Support** tab.

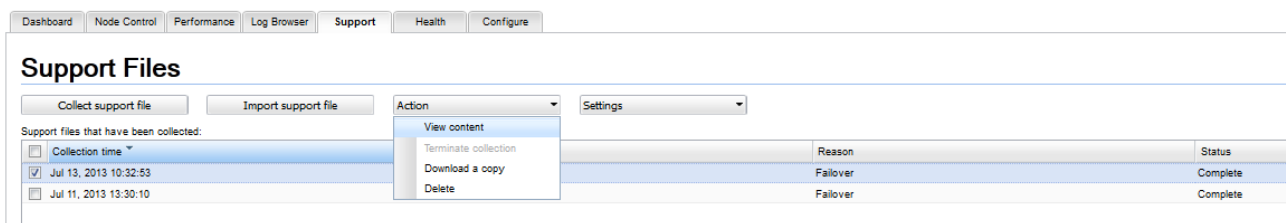
The **Support Files** screen displays.

2. Select a support file to view.

- a. In the row containing the support file you want to view (in the **User actions** column) click the **Actions** button.

- b. Select **Download a copy**.

- c. Specify where to save the file on your system, or you may choose to open and view the file directly.



5.2.3 Delete a support file

Use the delete file feature to delete a selected support file.

To delete a support file:

1. In CSSM, click the **Support** tab.

The Support Files screen displays.

2. Click to select a support file.

- a. In the row containing the support file you want 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.

The screenshot shows the 'Support Files' section of a management console. At the top, there are navigation tabs: Node Control, Performance, Log Browser, Support (selected), Terminal, Dashboard, Health, and Configure. Below the tabs, there are buttons for 'Collect support file' and 'Import support file', along with 'Action' and 'Settings' dropdown menus. A status message reads 'Support files that have been collected: (refreshing...)'. The main area contains a table with the following data:

Collection time	Method	Reason	Status	User actions
Mar 10, 2013 14:34:19	Uploaded	Uploaded by user admin	Transmitting	
Mar 9, 2013 15:15:54	Auto	Failover	Complete	
Mar 8, 2013 19:50:28	Auto	Failover	Complete	
Mar 8, 2013 18:48:57	Uploaded	Unloaded by user admin	Transmitting	
Mar 8, 2013 05:34:11	Auto			
Mar 8, 2013 04:27:20	Uploaded			
Mar 8, 2013 04:24:03	Uploaded			
Mar 8, 2013 03:24:28	Auto			
Mar 8, 2013 01:19:09	Auto			
Mar 7, 2013 23:00:41	Auto			

A dialog box titled 'Question' is overlaid on the table, containing a question mark icon and the text: 'Are you sure you want to delete the support file of Mar 8, 2013 19:50:28?'. It has 'Yes' and 'No' buttons.

5.3 Viewing support files

When a support file exists, either automatically or from using methods described in the preceding subsections, you can view the contents on the **Support** tab. Click the **Actions** pull-down menu and select the **View content** option.

Fields on the contents screen are described below.

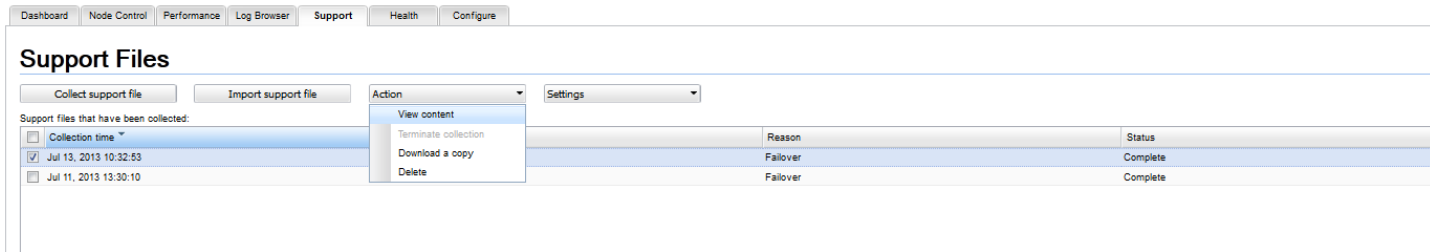
- **System Logs** (default tab) - Lists all the system logs for the cluster when the support file was created.
- **Node Information** - Lists information for all nodes in the cluster. See page 31.
- **Web Logs** - Lists all web logs for the cluster when the support file was created, for more details see page 32.
- **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. See page 33.

To view Sonexion support files:

1. In CSSM, click the **Support** tab.

The **Support Files** screen displays and lists 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 you want to view (in the **User actions** column) click the **Actions** button.
 - b. Select **View Content**.



The **Support Files** content screen opens and displays tabs that are discussed in the following paragraphs.

System Logs

The **System Logs** tab lists all the system logs for the cluster when the support file was created.

Support File Content Collection date : Jul 13, 2013 10:32:53
Reason : Failover [<< Back to the file list](#)

Viewing logs inside the support file

System Logs | Node Information | Web Logs | HA Logs | Application State

Filter (2617 rows)

Filter Name: no filter Save Remove

From: Hosts: Facility: Priority:
 To: PID: Program: Subsystem:
 Message: Apply Reset

Host	Facility	Priority	PID	Program	Subsystem	Date	Message
1 cstor00	local6	info	0	mgmt-ha		Jul 13, 2013 16:49:57	clbadm[118402]: info: Invoked: cibadmin -Ql
2 cstor00	local6	info	0	mgmt-ha		Jul 13, 2013 16:49:57	clbadm[118426]: info: Invoked: cibadmin -Ql
3 cstor02	cron	info	24212	CROND		Jul 13, 2013 16:50:01	(root) CMD (/usr/lib64/sa/sa1 -S DISK 1 1)
4 cstor03	cron	info	128883	CROND		Jul 13, 2013 16:50:01	(root) CMD (/usr/lib64/sa/sa1 -S DISK 1 1)
5 cstor00	cron	info	118786	CROND		Jul 13, 2013 16:50:01	(root) CMD (/usr/lib64/sa/sa1 -S DISK 1 1)
6 cstor01	cron	info	4207	CROND		Jul 13, 2013 16:50:01	(root) CMD (/usr/lib64/sa/sa1 -S DISK 1 1)
7 cstor01	daemon	info	0	dhcpd		Jul 13, 2013 16:50:10	DHCPDISCOVER from 00:50:cc:79:30:f7 via eth0
8 cstor01	daemon	info	0	dhcpd		Jul 13, 2013 16:50:10	DHCPDISCOVER on 172.16.255.105 to 00:50:cc:79:30:f7 via eth0
9 cstor01	daemon	info	0	dhcpd		Jul 13, 2013 16:50:13	DHCPDISCOVER from 00:50:cc:79:30:f7 via eth0
10 cstor01	daemon	info	0	dhcpd		Jul 13, 2013 16:50:13	DHCPDISCOVER on 172.16.255.105 to 00:50:cc:79:30:f7 via eth0
11 cstor01	local6	info	0	mgmt-ha		Jul 13, 2013 16:50:18	lrm[d]13530: info: rsc:prn-nfsd monitor[91] (pid 4240)
12 cstor01	local6	info	0	mgmt-ha		Jul 13, 2013 16:50:18	lrm[d]13530: info: operation monitor[91] on prn-nfsd for client 13533: pid 4240 exited with return code 0
13 cstor01	daemon	info	0	dhcpd		Jul 13, 2013 16:50:18	DHCPDISCOVER from 00:50:cc:79:30:f7 via eth0
14 cstor01	daemon	info	0	dhcpd		Jul 13, 2013 16:50:18	DHCPDISCOVER on 172.16.255.105 to 00:50:cc:79:30:f7 via eth0
15 cstor03	local6	info	129793	mdadm_conf_reg		Jul 13, 2013 16:50:28	[129797]: INFO: Running monitor
16 cstor03	local6	info	0	crm_resource		Jul 13, 2013 16:50:28	[129823]: info: Invoked: crm_resource -r cstor03_mdadm_conf_regenerate -g md5sum
17 cstor03	local6	info	0	crm_resource		Jul 13, 2013 16:50:29	[129850]: info: Invoked: crm_resource -L
18 cstor03	local6	info	0	crm_resource		Jul 13, 2013 16:50:29	[129871]: info: Invoked: crm_resource -r cstor02_mdadm_conf_regenerate -g md5sum
19 cstor02	local6	info	25217	mdadm_conf_reg		Jul 13, 2013 16:50:29	[25221]: INFO: Running monitor
20 cstor02	local6	info	0	crm_resource		Jul 13, 2013 16:50:29	[25245]: info: Invoked: crm_resource -r cstor02_mdadm_conf_regenerate -g md5sum
21 cstor02	local6	info	0	crm_resource		Jul 13, 2013 16:50:30	[25272]: info: Invoked: crm_resource -L

For each syslog, the following information is provided:

- **Host** - The hostname field consists of the host name (as configured on the host itself) or the IP address.
- **Facility** - This identifies who submitted the message. There are a small number of facilities defined. The kernel, the mail subsystem, FTP server, are just some examples of recognized facilities.
- **Priority** - The source or facility that generates the syslog message also specifies the severity of the message.
- **PID** - Process ID.

- **Program** - Name of program or module that produced the message.
- **Subsystem** - Filters Lustre, LustreError and some other classes of messages.
- **Date/Time** - The timestamp is the local time, in MMM DD HH:MM:SS format, of the device when the message was generated. This is also the default sort field, with the most recent collections at the top.
- **Message** - This is the text of the syslog message, along with some additional information about the process that generated the message.

Node information

The **Node** log displays version information for the selected node.

- **Node:** List of nodes available in this collection.

Click the down arrow and choose the node you wish to display, then click the **Apply** button.

- **Show:** Display the installed software, resources, and processes.

Click the down arrow and choose which module you wish to display, then click the **Apply** button.

Support File Content Collection date : Jul 13, 2013 10:32:53
Reason : Failover [<< Back to the file list](#)

Viewing logs inside the support file

System Logs **Node Information** Web Logs HA Logs Application State

Node : cstor00
Show : Installed software

Name	Version
OS	Scientific Linux release 6.2 (Carbon)
kernel_uname	Linux cstor00 2.6.32-220.7.1.el6.lustre.4109.x86_64 #1 SMP Mon Jun 24 18:03:23 PDT 2013 x86_64 x
puppetlabs-stdlib	2.3.2-0xrtx1.noarch
ibutils-libs	1.5.7-4.el6.x86_64
trinity-hw-profiles	1.4.0-17734.noarch
nfs-utils-lib	1.1.5-4.el6.x86_64
perl-String-CRC32	1.4-9.el6.x86_64
baseasystem	10.0-4.el6.noarch
symlinks	1.4-2.1.el6.x86_64
ppp4nagios	0.6.7-1.x86_64
drvf-w-seagate_mantaray-xrt2	1-1.xrtx.noarch
libxob	1.5-1.el6.x86_64
perl-TimeDate	1.16-11.1.el6.noarch
drvf-w-seagate_firefly-xrt6	1-3.xrtx.noarch
xytools-t10dif	1.0-5014.3937.x86_64
perl-NetAddr-IP	4.027-7.el6.x86_64
kernel-firmware	2.6.32-220.7.1.el6.lustre.4109.x86_64
libdbi	0.8.3-3.1.el6.x86_64
keyutils	1.4-3.el6.x86_64
ncurses-base	5.7-3.20090208.el6.x86_64
smp_utils	0.94-4.el6.x86_64
liblvm2dat	1.2.7.1-1.el6.x86_64

Web logs

Lists all web logs for the cluster when the support file was created. When a specific log is selected (using the Log File menu), the log entries are displayed in the lower pane. The following web logs are available:

- **wsgi_access.log**

These are the logs for the web server gateway interface.

- **access.log**

This is a copy of the access log.

- **error.log**

The error log contains information indicating when mysqld was started and stopped and also any critical errors that occur while the server is running.

Support File Content

Collection date : Jul 13, 2013 10:32:53
Reason : Failover

[<< Back to the file list](#)

Viewing logs inside the support file

System Logs
Node Information
Web Logs
HA Logs
Application State

Log file : wsgi_access.log

Viewing 100 of 172 total lines of content. To see all, [download a support file](#).

```

172.16.2.4 - t0inventory [11/Jul/2013:13:20:33 -0700] "POST /wsgi/inventory" - 200 -
172.16.2.1 - t0inventory [11/Jul/2013:13:20:34 -0700] "POST /wsgi/inventory" - 200 -
172.16.2.5 - - [11/Jul/2013:13:26:40 -0700] "GET /wsgi/restful/ClusterInfo" - 200 168
172.16.2.5 - t0inventory [11/Jul/2013:13:26:53 -0700] "POST /wsgi/inventory" - 200 -
172.16.2.6 - - [11/Jul/2013:13:28:08 -0700] "GET /wsgi/restful/ClusterInfo" - 200 168
172.16.2.6 - t0inventory [11/Jul/2013:13:28:19 -0700] "POST /wsgi/inventory" - 200 -
10.106.96.90 - admin [11/Jul/2013:13:32:44 -0700] "POST /wsgi/UserService" urn:getUserInfo 200 653
10.106.96.90 - admin [11/Jul/2013:13:32:45 -0700] "POST /wsgi/UserService" urn:getGroupInfo 200 1084
10.106.96.90 - admin [11/Jul/2013:13:32:45 -0700] "POST /wsgi/PluginService" urn:getPluginInfo 200 2318
10.106.96.90 - admin [11/Jul/2013:13:32:47 -0700] "POST /wsgi/CustomerWizardService" urn:getWizardClusterInfo 200 720
10.106.96.90 - admin [11/Jul/2013:13:32:47 -0700] "POST /wsgi/PropertyService" urn:getProperties 200 8855
10.106.96.90 - admin [11/Jul/2013:13:32:47 -0700] "POST /wsgi/ChangeDefaultPasswordService" urn:adminPasswordInitAllowed 200 527
10.106.96.90 - admin [11/Jul/2013:13:32:48 -0700] "POST /wsgi/VersionService" urn:getVersionInfo 200 632
10.106.96.90 - admin [11/Jul/2013:13:32:48 -0700] "POST /wsgi/UserService" urn:getServerInfo 200 746
10.106.96.90 - admin [11/Jul/2013:13:32:47 -0700] "GET /wsgi/async_notify/feed_latest/ALERT/20" - 200 801
10.106.96.90 - admin [11/Jul/2013:13:32:48 -0700] "GET /wsgi/async_notify/feed_long/any/600" - 200 1136
10.106.96.90 - admin [11/Jul/2013:13:32:53 -0700] "GET /wsgi/async_notify/feed_long/any/600/21" - 200 895
10.106.96.90 - admin [11/Jul/2013:13:32:53 -0700] "POST /wsgi/ChangeDefaultPasswordService" urn:changeDefaultPassword 200 858
10.106.96.90 - admin [11/Jul/2013:13:32:53 -0700] "GET /wsgi/async_notify/feed_long/any/600/22" - 200 478
10.106.96.90 - admin [11/Jul/2013:13:32:54 -0700] "GET /wsgi/async_notify/feed_long/any/600/23" - 200 586
10.106.96.90 - admin [11/Jul/2013:13:33:31 -0700] "GET /wsgi/async_notify/feed_long/any/600/24" - 200 895
10.106.96.90 - admin [11/Jul/2013:13:33:36 -0700] "GET /wsgi/async_notify/feed_long/any/600/25" - 200 1090
10.106.96.90 - admin [11/Jul/2013:13:33:36 -0700] "POST /wsgi/CustomerWizardActionService" urn:startWizardProcess 200 878
10.106.96.90 - admin [11/Jul/2013:13:33:36 -0700] "GET /wsgi/async_notify/feed_long/any/600/27" - 200 879
10.106.96.90 - admin [11/Jul/2013:13:33:36 -0700] "POST /wsgi/CustomerWizardService" urn:getWizardProcessInfo 200 701
10.106.96.90 - admin [11/Jul/2013:13:33:37 -0700] "POST /wsgi/CustomerWizardService" urn:getSSLUploadURL 200 539
10.106.96.90 - admin [11/Jul/2013:13:33:37 -0700] "POST /wsgi/CustomerWizardService" urn:getRoutesConfigUploadURL 200 564
10.106.96.90 - admin [11/Jul/2013:13:33:38 -0700] "POST /wsgi/CustomerWizardService" urn:getLnetConfigUploadURL 200 560
10.106.96.90 - admin [11/Jul/2013:13:33:38 -0700] "POST /wsgi/CustomerWizardService" urn:getLnetConfigUploadURL 200 560
10.106.96.90 - admin [11/Jul/2013:13:33:38 -0700] "POST /wsgi/CustomerWizardService" urn:getLnetConfigUploadURL 200 560

```

HA logs

Heartbeat is a daemon that provides cluster infrastructure (communication and membership) services to its clients. This allows clients to know about the presence (or absence) of peer processes on other machines and easily exchange messages with them.

Each pair of nodes has a shared HA configuration, which means that, if something goes wrong on any node in the pair the other node takes over all of services such as the lustre targets, Linux services, etc.

The logging feature used by HA writes down HA changes (resource transition, service restart, nodes being unresponsive) into log files named **ha-local.log** (for MGMT nodes) or **ha.log** (for all other nodes). All of these logs are stored in the **/var/log** dir and are linked to a shared storage providing the benefit that if one of the MGMT nodes' goes down the logs will not get lost. They can be accessed from the other node.

Support File Content

Collection date : Jul 13, 2013 10:32:53

Reason : Failover

[<< Back to the file list](#)

Viewing logs inside the support file

System Logs Node Information Web Logs HA Logs Application State

Log file: ha-local.log-20130713.gz

Viewing 100 of 2773 total lines of content. To see all, [download a support file](#).

```

Jul 12 00:01:04 cstor00 logd: [3339]: info: SIGHUP received, re-opened log files
Jul 12 00:02:36 cstor00 mdadm_conf_regenerate[53204]: [53208]: INFO: Running monitor
Jul 12 00:02:36 cstor00 crm_resource: [53234]: info: Invoked: crm_resource -r cstor00_mdadm_conf_regenerate -g md5sum
Jul 12 00:02:36 cstor00 crm_resource: [53260]: info: Invoked: crm_resource -L
Jul 12 00:02:36 cstor00 crm_resource: [53279]: info: Invoked: crm_resource -r cstor01_mdadm_conf_regenerate -g md5sum
Jul 12 00:04:21 cstor00 cibadmin: [54779]: info: Invoked: cibadmin -Q1
Jul 12 00:04:21 cstor00 cibadmin: [54810]: info: Invoked: cibadmin -Q1
Jul 12 00:06:22 cstor00 cibadmin: [56888]: info: Invoked: cibadmin --query --local --xpath //*/primitive[@id='cstor00_md64-fsys']
Jul 12 00:06:22 cstor00 cibadmin: [56903]: info: Invoked: cibadmin --query --local --xpath //*/primitive[@id='cstor00_md64-fsys']
Jul 12 00:09:02 cstor00 cib: [15139]: info: cib_stats: Processed 21 operations (13333.00us average, 0% utilization) in the last 10min
Jul 12 00:09:21 cstor00 cibadmin: [59484]: info: Invoked: cibadmin -Q1
Jul 12 00:09:21 cstor00 cibadmin: [59506]: info: Invoked: cibadmin -Q1
Jul 12 00:10:36 cstor00 mdadm_conf_regenerate[60883]: [60887]: INFO: Running monitor
Jul 12 00:10:36 cstor00 crm_resource: [60911]: info: Invoked: crm_resource -r cstor00_mdadm_conf_regenerate -g md5sum
Jul 12 00:10:36 cstor00 crm_resource: [60940]: info: Invoked: crm_resource -L
Jul 12 00:10:37 cstor00 crm_resource: [60961]: info: Invoked: crm_resource -r cstor01_mdadm_conf_regenerate -g md5sum
Jul 12 00:14:21 cstor00 cibadmin: [64044]: info: Invoked: cibadmin -Q1
Jul 12 00:14:22 cstor00 cibadmin: [64081]: info: Invoked: cibadmin -Q1
Jul 12 00:16:22 cstor00 cibadmin: [66223]: info: Invoked: cibadmin --query --local --xpath //*/primitive[@id='cstor00_md64-fsys']
Jul 12 00:16:22 cstor00 cibadmin: [66238]: info: Invoked: cibadmin --query --local --xpath //*/primitive[@id='cstor00_md64-fsys']
Jul 12 00:18:37 cstor00 mdadm_conf_regenerate[67927]: [67931]: INFO: Running monitor
Jul 12 00:18:37 cstor00 crm_resource: [67955]: info: Invoked: crm_resource -r cstor00_mdadm_conf_regenerate -g md5sum
Jul 12 00:18:37 cstor00 crm_resource: [67981]: info: Invoked: crm_resource -L
Jul 12 00:18:37 cstor00 crm_resource: [68000]: info: Invoked: crm_resource -r cstor01_mdadm_conf_regenerate -g md5sum
Jul 12 00:19:02 cstor00 cib: [15139]: info: cib_stats: Processed 24 operations (16666.00us average, 0% utilization) in the last 10min
Jul 12 00:19:21 cstor00 cibadmin: [68892]: info: Invoked: cibadmin -Q1
Jul 12 00:19:22 cstor00 cibadmin: [68914]: info: Invoked: cibadmin -Q1
Jul 12 00:24:21 cstor00 cibadmin: [73359]: info: Invoked: cibadmin -Q1
Jul 12 00:24:22 cstor00 cibadmin: [73393]: info: Invoked: cibadmin -Q1
Jul 12 00:26:22 cstor00 cibadmin: [75472]: info: Invoked: cibadmin --query --local --xpath //*/primitive[@id='cstor00_md64-fsys']

```

Application state

Application State shows the internal contents of the database which is used to track the states of the management application. This information is presented only for transparency, so that you can review in full the information that is being transmitted to the support staff. It is not intended for you to understand this information and you should not use it, as it may change format from version to version. However, it is information that will help the support staff to understand the context of the issue you are reporting, and may help them to debug the issue or identify unusual circumstances in which the issue appeared.

Support File Content

Collection date : Jul 13, 2013 10:32:53
 Reason : Failover

[<< Back to the file list](#)

Viewing logs inside the support file

System Logs
Node Information
Web Logs
HA Logs
Application State

Viewing 100 of 15287 total lines of content. To see all, [download a support file](#).

```

-- MySQL dump 10.13  Distrib 5.6.11, for Linux (x86_64)
--
-- Host: 127.0.0.1  Database: t0db
-----
-- Server version      5.6.11

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0 */;
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;

--
-- Table structure for table `be_command_log`
--

DROP TABLE IF EXISTS `be_command_log`;
/*!40101 SET @saved_cs_client      = @@character_set_client */;
/*!40101 SET character_set_client = utf8 */;
CREATE TABLE `be_command_log` (
  `command_id` int(11) NOT NULL AUTO_INCREMENT,
  `command` varchar(255) NOT NULL,
  `arguments` text NOT NULL,
  `exit_code` int(11) NOT NULL,
  `stdout` text,
  `stderr` text,
  `run_time` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP.

```

5.4 Use CSCLI for support bundles

Support bundles can be created using the `cscli support_bundles` command, documented on page **Error! Bookmark not defined.**

To collect a support bundle manually using CLI commands:

1. Log into the primary MGMT node via SSH. Run:

```
$ ssh -l admin primary_MGMT_node
```

2. Change to root user. Run:

```
$ sudo su -
```

3. Collect the support bundle.

- To collect the bundle using the default 45 minute time period, run:

```
[root@n000]# cscli support_bundle -c
```

Sample output:

```
[root@snx11000n000 ~]# cscli support_bundle -c
```

```
Collecting support bundle: id:4, nodes:all, time-window:45
minute(s)
```

- To collect the bundle with a different time period, run:

```
[root@n000]# cscli support_bundle -c -t minutes
```

Sample command and output:

```
[root@snx11000n000 ~]# cscli support_bundle -c -t 90
Collecting support bundle: id:4, nodes:all, time-window:90
minute(s)
```

4. To check the status of the data collection, enter the following:

```
[root@snx11000n000 ~]# cscli support_bundle -e 22
support_bundle: Error: Collection of support bundle with id 22 is in
progress
```

5. To export the support bundle, enter the following:

```
[root@snx11000n000 ~]# cscli support_bundle -e 22
Support bundle with id 22 saved in file support_bundle_2013-08-
08_10-54-07_310920.tgz
```

5.5 Interpret Sonexion support bundles

This section contains an overview of the support bundle contents. 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.

5.5.1 System-wide logs

- **lbug_syslog.csv**

This file contains syslog messages, in comma-separated value (CSV) format.

NOTE: The following files are not intended for use by Sonexion end users, but they may be valuable to Cray personnel and OEMs to better understand system states and behavior.

- **logs/access.log**

This log contains Apache HTTP access data.

- **logs/data_tables.sql**

This log 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**

This file contains the Apache error log.

- **logs/wsgi_access.log**

This mod_wsgi access log contains records of web service calls made from the CSSM.

5.5.2 Node-specific Logs:

- **nodes/nodename/conman.log**

This 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**

This 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**

This log contains a dump of kernel messages collected from the node.

- **nodes/nodename/fru_dump.yaml**

This file 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**

This 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**

This log contains state data of the MDRAID arrays, i.e., content of the `/proc/mdstat` file.

- **nodes/nodename/processes.csv**

This file 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**

This 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.csv**

This file contains a list of all installed packages with version information (`rpm -qa` output).

- **`nodes/nodename/states.csv`**

This file contains miscellaneous state data, including power, memory, uptime, CPU load, and Lustre targets.

6. Troubleshooting

6.1 CSSM Issues

6.1.1 GUI shows an Internal Error with a Message Similar to "All stages complete but Cluster is not in deployed State"

Problem description: This problem is most likely caused by an interruption in the operation of the customer wizard (e.g., the nodes were not down). The database contains unexpected values for this stage and progress for the cluster, so the customer wizard cannot continue.

Workaround: Return the Sonexion to pre-ship mode.

1. Log into the primary MGMT node via SSH.

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

2. Return the Sonexion to pre-ship mode. Run:

```
[admin@n000]$ /opt/xyratex/bin/cscli cluster_mode --mode pre-shipment
```

NOTE: The `/opt/xyratex/bin/` part of the command is not necessary. The command would be:

```
[admin@n000]$ cscli cluster_mode --mode pre-shipment
```

6.1.2 Upgrade is failing with "Metadata file does not match checksum" Errors

Problem description: There is a bug in the implementation of the `prepare_updates` routine that makes packages cache unusable.

Workaround: Run the following commands:

```
[MGMT]# rm -rf /var/cache/yum/*
[MGMT]# rm -rf /var/cache/yum_image/*
[MGMT]# createrepo /opt/xyratex/release/new_version_you_are_upgrading_to/
```

Where *new_version* is the new version you are upgrading to.

6.1.3 An Unclean Management Node Shutdown causes Database Corruption

Problem description: If the management node hosting the MySQL server is shut down uncleanly, including by a `stonith` command, the `t0db` database (named `filesystem_filesystem_name`) on the management nodes can become corrupt. This can manifest in several ways, including out-of-date information in the performance tab and problems assessing the management database `t0db`. There will usually be errors in the file `/var/log/mysqld.log` indicating the corrupt tables:

```
130220 8:20:28 [ERROR] /usr/libexec/mysqld: Table './filesystem_snx11003/MDS_OPS_DATA'
  is marked as crashed and should be repaired
130220 8:20:43 [ERROR] /usr/libexec/mysqld: Table './filesystem_snx11003/MDS_OPS_DATA'
  is marked as crashed and should be repaired
130220 8:20:58 [ERROR] /usr/libexec/mysqld: Table './filesystem_snx11003/MDS_OPS_DATA'
  is marked as crashed and should be repaired
```

Workaround: To repair the corrupted tables, execute the following procedure. In order to assist in troubleshooting if the procedure fails, please use the script command to save the output..

NOTE: For very large tables, repair operations can create temporary files that are larger than the available filesystem space. It is recommended that the available space be monitored during this procedure using the `df -h` command.

3. Log into the primary MGMT node via SSH. Run:

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

4. Sudo to root. Run:

```
[root@n000]# sudo su -
```

5. Using script, begin collecting program output. Run:

```
[root@n000]# script /tmp/t0db_repair_log
```

6. Shut down the mysql service. Run:

```
[root@n000]# service mysqld stop
```

7. Check all tables in t0db. Run:

```
[root@n000]# mysqlcheck filesystem_ filesystem_name
```

Sample output:

```
[root@havantfae00 filesystem_fs1]# mysqlcheck
  filesystem_ lustre_filesystem_name
filesystem_fs1.EVENT_DATA OK
filesystem_fs1.EVENT_INFO OK
filesystem_fs1.FILESYSTEM_AGGREGATE_DAY
error : Size of indexfile is: 15360 Should be: 18432
error : Corrupt
filesystem_fs1.FILESYSTEM_AGGREGATE_HOUR
error : Size of indexfile is: 224256 Should be: 251600
error : Corrupt
filesystem_fs1.FILESYSTEM_AGGREGATE_MONTH OK
filesystem_fs1.FILESYSTEM_AGGREGATE_WEEK OK
filesystem_fs1.FILESYSTEM_AGGREGATE_YEAR OK
filesystem_fs1.FILESYSTEM_INFO OK
```

Additional output omitted

8. Repair all tables. Run:

```
[root@n000]# mysqlcheck -s -r filesystem_name_fs1
```

Sample output:

```
[root@havantfae00 filesystem_fs1]# mysqlcheck -s -r filesystem_fs1
[root@havantfae00 filesystem_fs1]#
```

9. Verify that repair worked and that all tables are OK:

```
[root@n000]# mysqlcheck filesystem_fs1
```

Sample output:

```
[root@havantfae00 filesystem_fs1]# mysqlcheck filesystem_fs1
filesystem_fs1.EVENT_DATA OK
filesystem_fs1.EVENT_INFO OK
filesystem_fs1.FILESYSTEM_AGGREGATE_DAY OK
filesystem_fs1.FILESYSTEM_AGGREGATE_HOUR OK
filesystem_fs1.FILESYSTEM_AGGREGATE_MONTH OK
filesystem_fs1.FILESYSTEM_AGGREGATE_WEEK OK
filesystem_fs1.FILESYSTEM_AGGREGATE_YEAR OK
filesystem_fs1.FILESYSTEM_INFO OK
```

Additional output omitted.

10. Restart mysqld:

```
[root@n000]# service mysqld start
```

11. Exit from the script data collection:

```
[root@n000]# exit
```

This will save the output to the log file specified in Step 3.

If you encounter problems during the above procedure, or if problems persist on the system after the above procedure is finished, contact Cray support.

6.2 Networking Issues

6.2.1 Capturing Register Dumps for Mellanox Host Channel Adaptors and Switches.

Problem description: This procedure is used to capture register dumps from Mellanox Host Channel Adaptors (HCAs) and switches. The information in these dumps is the only effective method of debugging a failed switch or Mellanox HCA problems. Once the dump is captured it can then be attached to the JIRA ticket.

Workaround: To capture the register:

1. If debugging an HCA failure, log into the node with the failed HCA; if debugging a switch, log into a node that is still connected to the switch. Run:

```
[Client]$ ssh NODE
```

Where *NODE* is the node with the failed HCA or the node that this still connected to the switch.

2. Sudo to root. Run:

```
[NODE]$ sudo su -
```

3. Start MST. Run:

```
[NODE]# mst start
```

4. Discover the network. Run:

```
[NODE]# mst ib add
```

This populates `/dev/mst/` with IB devices, including the faulted switch or HCA.

5. Capture the dump. Run:

```
[NODE]# mstdump /dev/mst/device_name > mstdump.out
```

Where *device_name* is the name of the faulted switch or HCA. Switch names start with "SW", and HCAs with "CA".

6. Attach the `mstdump.out` file to the relevant JIRA ticket.

6.3 RAID/HA Issues

6.3.1 RAIDs are not assembled correctly on the Nodes

Problem description: When an MDRAID device fails (for example, as a result of an expander reset temporarily removing several disks) the STONITH resource detects this change within its monitoring interval (10 minutes) and attempts to reassemble the MDRAID device on its OSS node. If the MDRAID device does not rebuild successfully, then the reassembly attempt times out after three minutes and the STONITH resource records a “failed actions” message for the OSS node.

The STONITH resource then tries to assemble the MDRAID device on the OSS node's HA partner node. If the rebuild is not successful on the HA partner node, then the reassembly attempt times out after three minutes and the STONITH resource records another “failed actions” message for the HA partner node.

After these failed attempts, the STONITH resource no longer tries to assemble the -RAID resource but leaves the first three resources in the group assembled.

IMPORTANT: GridRAID systems running release 1.4 requires SU 11 to be installed before attempting this procedure.

Workaround: Use the following steps to manually recover the RAID. This procedure assumes that onsite personnel have identified the OSS node(s) that control the failed RAID array(s). Please note that:

- Even numbered OSS nodes natively control even numbered MDRAID devices (md0, md2, md4, and md6).
- Odd numbered OSS nodes natively control odd numbered MDRAID devices (md1, md3, md5, and md7).
- If a native OSS node is in a failover state, control of the MDRAID devices that it natively controls will migrate to its HA partner node. It is possible to recover the MDRAID device using either of these HA partner OSS nodes.

In the Sonexion solution, an enclosure and two controllers are bundled in the modular SSU. Each controller hosts one OSS node; there are two OSS nodes per SSU. Within an SSU, the OSS nodes are organized in a High Availability (HA) pair with sequential numbers (for example, node04 / node05 or lmtest704 / lmtest705). If an OSS node goes down because its controller fails, its resources migrate to the HA partner/OSS node in the other controller.

The 84 disk drives in a Sonexion SSU are configured as:

- 8 OSTs, each a RAID6 array consisting of 8 data disks and 2 parity disks
- 2 SSDs partitioned to create multiple independent RAID1 slices, used for MDRAID write intent bitmaps and external OST/liskfs file system journals
- 2 hot spares

The virtual drives defined by the RAID6 arrays are referred to as MDRAID devices, numbered sequentially from 0 through 7, for example, md0. Within the STONITH resource, there are resources defined for each MDRAID device, used by the STONITH resource to control the MDRAID device. For example, lmtest704_md0-raid is the resource that controls the MDRAID device md0.

When an MDRAID device fails (for example, as a result of an expander reset removing several disks) the STONITH resource detects this change within its monitoring interval (10 minutes) and attempts to reassemble the MDRAID device on its OSS node. If the MDRAID device does not rebuild successfully, then the reassembly attempt times out after three minutes and the STONITH resource records a failed actions message for the OSS node.

The STONITH resource then tries to assemble the MDRAID device on the OSS node's HA partner node. If the rebuild is not successful on the HA partner node, then the reassembly attempt times out after three minutes and the STONITH resource records another failed actions message for the HA partner node.

After these failed attempts, the STONITH resource no longer tries to assemble the RAID resource but leaves the first three resources in the group assembled.

6.2.1 Preparing to Recover a Failed RAID Array

To prepare to recover a failed RAID Array:

1. Log into the primary MGMT node via SSH. Run:

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

2. Change to root user. Run:

```
[admin@n000]$ sudo su -
```

3. Determine if either of the OSS nodes that control the failed MDRAID device are offline. If so, power on the downed OSS node(s). On the primary MGMT node, run:

```
[root@n000]# pm -l OSS_nodename
```

Here is sample output:

```
[root@snx11000n000 ~]# pm -l snx11000n004
Command completed successfully
```

If both OSS nodes are down, repeat Step 5 on the HA partner node.

4. Wait several minutes, and then log into the previously downed OSS node via SSH to verify that it is back online. Run:

```
[root@n000]# ssh OSS_nodename
```

Here is sample output:

```
[root@snx11000n000 ~]# ssh snx11000n004
[root@snx11000n004 ~]#
```

If both OSS nodes were down, repeat Step 4 on the HA partner node.

5. Log into the OSS node that should natively (i.e. even/odd MDs to even/odd OSSs) control the MDRAID with a problem, via SSH. Run:

```
[OSS node]# ssh OSS_nodename
```

6. Check `/var/log/messages` to see if there was a report of an invalid mdraid superblock:

```
md: sdXXdoes not have a valid v1.2 superblock, not importing!
```

where `sdXX` is the drive with the bad superblock. If this error message is present, it is necessary to zero out the device superblock:

```
[OSS node]# mdadm -zero-superblock /dev/sdXX
```

7. Use the `crm_mon` utility to verify that a failed actions message was recorded for the failed MDRAID device. Also verify that the first three resources in the failed MDRAID device's resource group have failed over to the HA partner node. Run:

```
[OSS node]# crm_mon -lr
```

IMPORTANT: When reviewing the `crm_mon` output, note the failed MDRAID device's resource group name. You will need this information when performing the procedure to recover the failed RAID array.

Here is sample output showing the resource groups and the failed actions messages (as highlighted):

```
[root@snx11000n004 ~]# crm_mon -lr
=====
Last updated: Wed Jan 23 17:30:10 2013
Last change: Wed Jan 23 17:16:30 2013 via cibadmin on snx11000n005
Stack: Heartbeat
Current DC: snx11000n004 (8ab209a5-874a-404d-af1c-1afa84cc18a9) - partition with quorum
Version: 1.1.6.1-2.el6-0c7312c689715e096b716419e2ebc12b57962052
2 Nodes configured, unknown expected votes
55 Resources configured.
Online: [ snx11000n004 snx11000n005 ]
snx11000n004-stonith (stonith:external/gem_stonith): Started snx11000n004
snx11000n005-stonith (stonith:external/gem_stonith): Started snx11000n005
snx11000n004_mdadm_conf_regenerate (ocf::heartbeat:mdadm_conf_regenerate):Started
snx11000n004
snx11000n005_mdadm_conf_regenerate (ocf::heartbeat:mdadm_conf_regenerate):Started
snx11000n005
baton (ocf::heartbeat:baton): Started snx11000n005
snx11000n004_ibstat (ocf::heartbeat:ibstat): Started snx11000n004
snx11000n005_ibstat (ocf::heartbeat:ibstat): Started snx11000n005
Resource Group: snx11000n004_md0-group
snx11000n004_md0-wibr (ocf::heartbeat:XYRAID): Started snx11000n004
snx11000n004_md0-jnlr (ocf::heartbeat:XYRAID): Started snx11000n004
snx11000n004_md0-wibs (ocf::heartbeat:XYMNTR): Started snx11000n004
snx11000n004_md0-raid (ocf::heartbeat:XYRAID): Stopped
snx11000n004_md0-fsys (ocf::heartbeat:XYMNTR): Stopped
snx11000n004_md0-stop (ocf::heartbeat:XYSTOP): Stopped
Resource Group: snx11000n004_md1-group
snx11000n004_md1-wibr (ocf::heartbeat:XYRAID): Started snx11000n005
snx11000n004_md1-jnlr (ocf::heartbeat:XYRAID): Started snx11000n005
```

```

snx11000n004_md1-wibs (ocf::heartbeat:XYMNTR): Started snx11000n005
snx11000n004_md1-raid (ocf::heartbeat:XYRAID): Started snx11000n005
snx11000n004_md1-fsys (ocf::heartbeat:XYMNTR): Started snx11000n005
snx11000n004_md1-stop (ocf::heartbeat:XYSTOP): Started snx11000n005
Resource Group: snx11000n004_md2-group
snx11000n004_md2-wibr (ocf::heartbeat:XYRAID): Started snx11000n004
snx11000n004_md2-jnlr (ocf::heartbeat:XYRAID): Started snx11000n004
snx11000n004_md2-wibs (ocf::heartbeat:XYMNTR): Started snx11000n004
snx11000n004_md2-raid (ocf::heartbeat:XYRAID): Started snx11000n004
snx11000n004_md2-fsys (ocf::heartbeat:XYMNTR): Started snx11000n004
snx11000n004_md2-stop (ocf::heartbeat:XYSTOP): Started snx11000n004
Resource Group: snx11000n004_md3-group
snx11000n004_md3-wibr (ocf::heartbeat:XYRAID): Started snx11000n005
snx11000n004_md3-jnlr (ocf::heartbeat:XYRAID): Started snx11000n005
snx11000n004_md3-wibs (ocf::heartbeat:XYMNTR): Started snx11000n005
snx11000n004_md3-raid (ocf::heartbeat:XYRAID): Started snx11000n005
snx11000n004_md3-fsys (ocf::heartbeat:XYMNTR): Started snx11000n005
snx11000n004_md3-stop (ocf::heartbeat:XYSTOP): Started snx11000n005
Resource Group: snx11000n004_md4-group
snx11000n004_md4-wibr (ocf::heartbeat:XYRAID): Started snx11000n004
snx11000n004_md4-jnlr (ocf::heartbeat:XYRAID): Started snx11000n004
snx11000n004_md4-wibs (ocf::heartbeat:XYMNTR): Started snx11000n004
snx11000n004_md4-raid (ocf::heartbeat:XYRAID): Started snx11000n004
snx11000n004_md4-fsys (ocf::heartbeat:XYMNTR): Started snx11000n004
snx11000n004_md4-stop (ocf::heartbeat:XYSTOP): Started snx11000n004
Resource Group: snx11000n004_md5-group
snx11000n004_md5-wibr (ocf::heartbeat:XYRAID): Started snx11000n005
snx11000n004_md5-jnlr (ocf::heartbeat:XYRAID): Started snx11000n005
snx11000n004_md5-wibs (ocf::heartbeat:XYMNTR): Started snx11000n005
snx11000n004_md5-raid (ocf::heartbeat:XYRAID): Started snx11000n005
snx11000n004_md5-fsys (ocf::heartbeat:XYMNTR): Started snx11000n005
snx11000n004_md5-stop (ocf::heartbeat:XYSTOP): Started snx11000n005
Resource Group: snx11000n004_md6-group
snx11000n004_md6-wibr (ocf::heartbeat:XYRAID): Started snx11000n004
snx11000n004_md6-jnlr (ocf::heartbeat:XYRAID): Started snx11000n004
snx11000n004_md6-wibs (ocf::heartbeat:XYMNTR): Started snx11000n004
snx11000n004_md6-raid (ocf::heartbeat:XYRAID): Started snx11000n004
snx11000n004_md6-fsys (ocf::heartbeat:XYMNTR): Started snx11000n004
snx11000n004_md6-stop (ocf::heartbeat:XYSTOP): Started snx11000n004
Resource Group: snx11000n004_md7-group
snx11000n004_md7-wibr (ocf::heartbeat:XYRAID): Started snx11000n005
snx11000n004_md7-jnlr (ocf::heartbeat:XYRAID): Started snx11000n005
snx11000n004_md7-wibs (ocf::heartbeat:XYMNTR): Started snx11000n005
snx11000n004_md7-raid (ocf::heartbeat:XYRAID): Started snx11000n005
snx11000n004_md7-fsys (ocf::heartbeat:XYMNTR): Started snx11000n005
snx11000n004_md7-stop (ocf::heartbeat:XYSTOP): Started snx11000n005
Failed actions:
snx11000n004_md0-raid_start_0 (node=snx11000n005, call=134, rc=-2, status=Timed Out):
    unknown exec error
snx11000n004_md0-raid_start_0 (node=snx11000n004, call=134, rc=-2, status=Timed Out):
    unknown exec error

```

If the RAID fails to assemble and generates messages similar to those below, then the RAID recovery procedure has failed, go to Step 10:

```
mdadm: ignoring /dev/disk/by-id/wwn-0x5000cca01b3d4224 as it reports
/dev/disk/by-id/wwn-0x5000cca01b3cf13c as failed
mdadm: ignoring /dev/disk/by-id/wwn-0x5000cca01b3d7e24 as it reports
/dev/disk/by-id/wwn-0x5000cca01b3cf13c as failed
mdadm: ignoring /dev/disk/by-id/wwn-0x5000cca01b3d6080 as it reports
/dev/disk/by-id/wwn-0x5000cca01b3cf13c as failed
mdadm: ignoring /dev/disk/by-id/wwn-0x5000cca01c375f04 as it reports
/dev/disk/by-id/wwn-0x5000cca01b3cf13c as failed
```

If the above error messages do not appear, go to Step 11.

8. Abort this procedure and contact Cray support.

Cray support will require the mdraid superblock data to be collected in order to debug the problem. In release 1.4.0 and later, this information can be collected using the `mdraid-analyze` script located at `/usr/sbin/`. In release 1.3.1 or earlier, use the `collect_superblock.sh` script to collect this data.

- a. Download the `collect_superblock.sh` script from XIC to `/tmp` on the OSS node with the failed raid.
- b. Exit the OSS node. Run:

```
[OSS node]# exit
```

- c. Run the script:

```
[root@n000]# pdsh -g OSS_node /tmp/collect_superblock.sh
/var/lib/mdraidscripts/mdadm.conf
```

9. If the first three resources in the failed MDRAID device's resource group have failed over to the HA partner node, exit the OSS node and log into the HA partner node via SSH. Run:

```
[OSS node]# exit
[root@n000]# ssh HA_partner_nodename
```

The procedure to prepare for recovering a failed RAID array is now complete. Proceed to the next section for the procedure to recover the RAID array.

Recovering a failed RAID Array

This procedure describes how to force assemble an MDRAID device to recover a failed RAID array.

CAUTION: Assembling a RAID array with the `--force` argument can result in data loss or data corruption. This procedure should only be used as a last resort.

10. Log into the previously downed OSS node via SSH. Run:

```
[root@n000]# ssh OSS_nodename
```

11. Stop the resource group containing the failed MDRAID device. Run:

```
[OSS node]# stop_xyraid resource_group_name
```

Where *resource_group_name* is the resource group name identified in Section 9 of the previous procedure (**6.2.1 Preparing to Recover a Failed RAID Array**).

Here is sample output:

```
[root@snx11000n005 ~]# stop_xyraid snx11000n004_md0-group
[root@snx11000n005 ~]#
```

12. Use the `unmanage` command to allow the resource group resources to be started outside of the STONITH resource. Run:

```
[OSS node]# unmanage_xyraid resource_group_name
```

Here is sample output:

```
[root@snx11000n005 ~]# unmanage_xyraid snx11000n004_md0-group
[root@snx11000n005 ~]#
```

13. Clean the resource group to remove the failed actions. Run:

```
[OSS node]# clean_xyraid resource_group_name
```

Here is sample output:

```
[root@snx11000n005 ~]# clean_xyraid snx11000n004_md0-group
Cleaning up snx11000n004_md0-wibr on snx11000n004
Cleaning up snx11000n004_md0-wibr on snx11000n005
Cleaning up snx11000n004_md0-jnlr on snx11000n004
Cleaning up snx11000n004_md0-jnlr on snx11000n005
Cleaning up snx11000n004_md0-wibs on snx11000n004
Cleaning up snx11000n004_md0-wibs on snx11000n005
Cleaning up snx11000n004_md0-raid on snx11000n004
Cleaning up snx11000n004_md0-raid on snx11000n005
Cleaning up snx11000n004_md0-fsys on snx11000n004
Cleaning up snx11000n004_md0-fsys on snx11000n005
Cleaning up snx11000n004_md0-stop on snx11000n004
Cleaning up snx11000n004_md0-stop on snx11000n005
Waiting for 13 replies from the CRMd..... OK
```

14. If you determined in Section **6.2.1 Preparing to Recover a Failed RAID Array**, that the first three resources in the failed MDRAID device's resource group have failed over to the HA partner node, follow Steps 4.a and 4.b below:

- a. Log into the OSS node that natively controls the MDRAID device, via SSH. Run:

```
[OSS node]# ssh OSS_nodename
```

- b. Fail back resources to the OSS node. Run:

```
[OSS node]# failback_xyraid
```

Here is sample output:

```
[root@snx11000n004 ~]# failback_xyraid
[root@snx11000n004 ~]#
```

15. Determine if the `--force` argument is necessary to assemble the MDRAID device. Run:

```
[OSS node]# mdraid-activate -d resource_group_name
```

Here is sample output showing an unsuccessful attempt to assemble the MDRAID device without the `--force` argument:

```
[root@snx11000n004 ~]# mdraid-activate -d snx11000n004_md0-group
mdadm: /dev/md/snx11000n004:md128 has been started with 2 drives.
mdadm: /dev/md/snx11000n004:md129 has been started with 2 drives.
mdadm: failed to RUN_ARRAY /dev/md/snx11000n004:md0: Input/output error
mdadm: Not enough devices to start the array.
mdadm: failed to RUN_ARRAY /dev/md/snx11000n004:md0: Input/output error
mdadm: Not enough devices to start the array.
mdadm: failed to RUN_ARRAY /dev/md/snx11000n004:md0: Input/output error
mdadm: Not enough devices to start the array.
mdadm: failed to RUN_ARRAY /dev/md/snx11000n004:md0: Input/output error
mdadm: Not enough devices to start the array.
mdadm: failed to RUN_ARRAY /dev/md/snx11000n004:md0: Input/output error
mdadm: Not enough devices to start the array.
mdadm: failed to RUN_ARRAY /dev/md/snx11000n004:md0: Input/output error
mdadm: Not enough devices to start the array.
mdadm: failed to RUN_ARRAY /dev/md/snx11000n004:md0: Input/output error
mdadm: Not enough devices to start the array.
mdadm: failed to RUN_ARRAY /dev/md/snx11000n004:md0: Input/output error
mdadm: Not enough devices to start the array.
mdadm: failed to RUN_ARRAY /dev/md/snx11000n004:md0: Input/output error
mdadm: Not enough devices to start the array.
mdraid-activate 358: unable to assemble snx11000n004:md0
```

NOTE: If the above assembly attempt was successful, proceed to Step 8.

16. Assemble the MDRAID device using the `--force` argument. If you are performing this procedure on a system running 1.2.0, 1.2.1 or 1.2.3, run the following command:

```
[OSS node]# mdraid-activate -df resource_group_name
```

If you are performing this procedure on a system running 1.3.1, 1.4.0 or 1.5.0, run the following command:

```
[OSS node]# mdraid-activate -f i_am_sure_i_want_to_do_this,exit -d
resource_group_name
```

Here is sample output of a successful 'forced' assembly of an MDRAID device on 1.2.0, 1.2.1 or 1.2.3:

```
[root@snx11000n004 ~]# mdraid-activate -df snx11000n004_md0-group
mdadm: /dev/md/snx11000n004:md129 has been started with 2 drives.
mdadm: forcing event count in /dev/disk/by-id/wwn-0x5000cca01c477720(3) from 3 upto 16
mdadm: forcing event count in /dev/disk/by-id/wwn-0x5000cca01c472818(8) from 3 upto 16
mdadm: forcing event count in /dev/disk/by-id/wwn-0x5000cca01b4a0ec4(9) from 3 upto 16
mdadm: clearing FAULTY flag for device 8 in /dev/md/snx11000n004:md0 for /dev/disk/by-id/wwn-0x5000cca01c477720
mdadm: clearing FAULTY flag for device 7 in /dev/md/snx11000n004:md0 for /dev/disk/by-id/wwn-0x5000cca01c472818
mdadm: clearing FAULTY flag for device 3 in /dev/md/snx11000n004:md0 for /dev/disk/by-id/wwn-0x5000cca01b4a0ec4
mdadm: Marking array /dev/md/snx11000n004:md0 as 'clean'
```



```
mdadm: /dev/md/snx11000n004:md0 has been started with 10 drives.
assembled snx11000n004:md0 in 1 tries
```

Here is sample output of a successful 'forced' assembly of an MDRAID device on 1.3.1:

```
[root@lmsse05 ~]# mdraid-activate -f i_am_sure_i_want_to_do_this,exit -d lmsse04_md0-
group
mdadm: /dev/md/lmsse04:md129 has been started with 2 drives.
mdadm: failed to RUN_ARRAY /dev/md/lmsse04:md0: Input/output error
mdadm: Not enough devices to start the array.
mdadm: failed to RUN_ARRAY /dev/md/lmsse04:md0: Input/output error
mdadm: Not enough devices to start the array.
/usr/lib/ocf/lib/heartbeat/xrtx-ocf-shellfuncs: line 867: ocf_log: command not found
mdadm: forcing event count in /dev/disk/by-id/wwn-0x5000c500212d3f2b(3) from 19 upto 39
mdadm: clearing FAULTY flag for device 7 in /dev/md/lmsse04:md0 for /dev/disk/by-id/wwn-
0x5000c500212d3f2b
mdadm: Marking array /dev/md/lmsse04:md0 as 'clean'
mdadm: /dev/md/lmsse04:md0 has been started with 8 drives (out of 10).
assembled lmsse04:md0 in 1 tries
[root@lmsse05 ~]
```

IMPORTANT: If the MDRAID device failed to assemble, stop and contact Cray Support.

17. Run the `e2fsck` command on the MDRAID device.

IMPORTANT: Do not run the `e2fsck` command on a 1.2.0 or 1.2.1 system with OSTs larger than 16 TB without first installing system upgrade 1.2.0-MRP-1 or system upgrade 1.2.1-MRP-1.

- To check whether the OST size is larger than 16TB, run:

```
[OSS node]# mdadm --misc --detail /dev/md3 | grep Array
```

Example output of the command run on a very small OST:

```
[root@mytest204 ~]# mdadm --misc --detail /dev/md3 | grep Array
Array Size : 125829120 (120.00 GiB 128.85 GB)
```

- To check the version of `e2fsck` on your system, run:

```
[OSS node]# e2fsck -V
```

The version to run if the OST is larger than 16TB should be at least 1.42.6.x1.

When you have the correct version of the `e2fsck` command, run:

```
[OSS node]# e2fsck -fp /dev/MDRAID_device
```

Here is sample output:

```
[root@snx11000n004 ~]# e2fsck -fp /dev/md0
testfs-OST0000: recovering journal
testfs-OST0000: 86/7879680 files (2.3% non-contiguous),
509811/31457280 blocks
```

18. Stop the MDRAID device. Run:

```
[OSS node]# mdraid-deactivate resource_group_name
```

Here is sample output:

```
[root@snx11000n004 ~]# mdraid-deactivate snx11000n004_md0-group
[root@snx11000n004 ~]#
```

19. Manage the MDRAID device's resource group. Run:

```
[OSS node]# manage_xyraid resource_group_name
```

Here is sample output:

```
[root@snx11000n004 ~]# manage_xyraid 04 ~]#
```

20. Start the MDRAID device's resource group. Run:

```
[OSS node]# start_xyraid -nowait resource_group_name
```

Here is sample output:

```
[root@snx11000n004 ~]# start_xyraid snx11000n004_md0-group
[root@snx11000n004 ~]#
```

21. Use the `crm_mon` utility to verify that the MDRAID device's resource group started correctly, which can take several minutes. Run:

```
[OSS node]# crm_mon -l
```

Here is sample output showing a healthy OST group:

```
[root@snx11000n004 ~]# crm_mon -l
===== 23 18:00:58 2013
Last change: Wed Jan 23 18:00:18 2013 via cibadmin on snx11000n004
Stack: Heartbeat
Current DC: snx11000n005 (8ab209a5-874a-404d-af1c-1afa84cc18a9) -
partition with quorum
Version: 1.1.6.1-2.el6-0c7312c689715e096b716419e2ebc12b57962052
2 Nodes configured, unknown expected votes
55 Resources configured.
Online: [ snx11000n004 snx11000n005 ]
snx11000n004-stonith (stonith:external/gem_stonith): Started snx11000n004
snx11000n005-stonith (stonith:external/gem_stonith): Started snx11000n005
snx11000n004_mdadm_conf_regenerate
(ocf::heartbeat:mdadm_conf_regenerate):Started snx11000n004
snx11000n005_mdadm_conf_regenerate
(ocf::heartbeat:mdadm_conf_regenerate):Started snx11000n005
baton (ocf::heartbeat:baton): Started snx11000n005
snx11000n004_ibstat (ocf::heartbeat:ibstat): Started snx11000n004
snx11000n005_ibstat (ocf::heartbeat:ibstat): Started snx11000n005
Resource Group: snx11000n004_md0-group
snx11000n004_md0-wibr (ocf::heartbeat:XYRAID): Started snx11000n004
snx11000n004_md0-jnlr (ocf::heartbeat:XYRAID): Started snx11000n004
snx11000n004_md0-wibs (ocf::heartbeat:XYMNTR): Started snx11000n004
snx11000n004_md0-raid (ocf::heartbeat:XYRAID): Started snx11000n004
snx11000n004_md0-fsys (ocf::heartbeat:XYMNTR): Started snx11000n004
snx11000n004_md0-stop (ocf::heartbeat:XYSTOP): Started snx11000n004
Resource Group: snx11000n004_md1-group
```


6.4 Other Issues

6.4.1 Non-Responsive Server

Problem description: A server node failed and is not responding to power manage commands to reboot.

Workaround: To revive the node, using conman, run the `ipmi` command to start the node.

1. Log into the primary MGMT node via SSH. Run:

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

2. Log in as root user. Run:

```
[admin@n000]$ sudo su -
```

3. Log in via console manager. Run:

```
[MGMT]# conman nodename-gem
```

4. Issue the following command. Run:

```
[GEM]# -ipmi_power 4
```

6.4.2 Attempting to connect to GEM with conman results in the message "Port Already In Use"

Problem Description: The conman daemon automatically connects to all the x86 CPU targets in its configuration file using BMC (Baseboard Management Controller) SOL (serial on LAN), and then uses these SOL connections to open UART connections to GEM. These SOL streams for x86 and GEM is always running in the background.

When the user opens a terminal to a node using conman, a terminal front end is being attached to background SOL stream. If the conman daemon is already running on one node and the daemon is started on the other, that will disrupt the open SOL stream on the first node.

This disruption will actually result in the 2nd node forcing the SOL connection on the 1st node to close. But this forced closure occurs before the GEM UART resources can be freed, thus resulting in the "Port Already In Use" message.

SOL doesn't support multiple connections, hence if the multiple sessions are started to x86 and GEM on SATI then conman service will fail / error / crash.

Workaround: If the "Port already in use" message is encountered the recommendation is to load conman with either the `-f` or `-j` options.

The `-f` flag specifies that write-access to the console should be "forced", thereby stealing the console away from existing clients having write privileges. The original clients are informed by conman of who perpetrated the theft as their connections are terminated.

The `-j` flag specifies that write-access to the console should be "joined", thereby sharing the console with existing clients having write privileges. The original clients are informed by `conman` that a new client has been granted write privileges.

If either of the above options does not allow `conman` to attach to GEM, then restart the `ser2net` service on the node in question. To restart the `ser2net` service, run:

```
[root@n000]# ssh [node_name]
[OSS node]# service ser2net restart
```

6.4.3 HA failover in the event of a `xyvinc` malfunction

Problem Description: A new OSS controller failed to come online after replacement. The `restore_ha_config.sh` script did not complete because the High-Availability (HA) system has multiple nodes defined with the same hostname.

Workaround: Manually remove the extraneous nodes and rerun the `restore_ha_config.sh` script.

1. Log into the new controller. Run:

```
[NEW_CONTROLLER]$ ssh -l admin new_controller
```

2. Log in as root user on the new controller. Run:

```
[NEW_CONTROLLER]$ sudo su-
```

3. Check to see if the HA system has multiple nodes defined with the same hostname.

Run:

```
[NEW_CONTROLLER]# crm_mon -lr
```

If the output has more than one entry with the new hostname, for example,

```
Online: [ tsesys2n05 tsesys2n04 ]
Offline: [ tsesys2n04 ]
```

and no resources are started on the new node, then it is necessary to remove the node that does not correspond to the working node.

4. Retrieve the `uuid` of the new controller. Run:

```
[NEW_CONTROLLER]# crm_uuid
```

For example:

```
[root@tsesys2n04 ~]# crm_uuid
e596dc72-84a9-4a61-8873-5cd728bea632
[root@tsesys2n04 ~]#
```

5. Edit the CIB and remove the entries for this hostname that do *not* match this `uuid`:

```
[NEW_CONTROLLER]# crm configure edit
```

This command puts the user in `'vi'` mode, editing the CIB. At the very top of the CIB are the node definitions, for example,

```
node $id="3334b519-897c-4438-bd65-2750e4094a3a" tsesys2n05
```

```
node $id="e596dc72-84a9-4a61-8873-5cd728bea632" tsesys2n04
node $id="e534b532-987e-54a3-8873-4ab8798ab1ba" tsesys2n04
```

Remove the entire line corresponding to the node that has the same hostname, but does **not** have the `uuid` (the second field) of the working node. In this example, since the `uuid` of the working `tsesys2n04` is `e596dc72-84a9-4a61-8873-5cd728bea632`, the third line must be removed.

6. Exit the edit session with the sequence `:wq`
7. Check to see if the new controller still has multiple nodes defined with the same hostname, it should now only show two online nodes. Run:

```
[NEW_CONTROLLER]# crm_mon -1
```

8. Stop the heartbeat service on the new controller. Run:

```
[NEW_CONTROLLER]# service heartbeat stop
```

9. Rerun the `restore_ha_config.sh` command:

```
[NEW_CONTROLLER]# /opt/xyratex/bin/restore_ha_config.sh
```

The `restore_ha_config.sh` command should now finish as normal.

7. CSCLI User Documentation

This chapter provides reference information for Sonexion's CLI command interface.

CLI commands are organized by mode; that is, certain commands are available according to 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
- Daily Mode

7.1 CSCLI Overview

7.1.1 Site Configuration mode

Site Configuration (custWizard) 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.

7.1.2 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.

7.1.3 Guest Mode

A “Guest” account lets non-privileged users run some commands to obtain information about system using read-only access to the system. Depending on the privileges, a subset of CSCLI commands are provided for a Guest account.

7.1.4 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, it reports an error 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, this stops the validation of upcoming arguments and raises an exception.

IMPORTANT: Exercise caution before using the `-y` or `--force` parameter. For all CSCLI commands, it is recommended that the `-y` and `--force` parameters be used very cautiously, even if they are required parameters.

7.2 Summary of Changes in Release 1.5.0

Table 2 shows commands that were added or changed in this release:

Table 2. CLI Changes for Release 1.5.0

No.	Addition/Change	CLI Command	Description	Component
1.	Added	cng	Exporter management command for CNG	CIFS/NFS
2.	Added	ibstat_check	Disables/Enables HA's probing of the high speed network	HA stack
3.	Added	manage_guest	Manages the guest account	User Management
4.	Added	show_node_versions	Changes the image for a diskless node	System upgrades
5.	Added	service_console	Enables or disables the service console or displays the state of the service console	Serviceability

No.	Addition/Change	CLI Command	Description	Component
6.	Added	lustre_users	Lustre Users command in Daily Mode. Many commands have changed and several commands have been added. These commands can now be performed from the Daily Mode. Refer to the New and Old Commands Comparison Table for a summary of the commands.	Advanced User Authentication
7.	Added support_bundle subcommand	support_bundle collect	Displays pdsh-style node names and the time window in minutes	Support bundle settings
		support_bundle show	Displays triggers that initiate automatic bundle collection, a list of support bundles collected and the purge limit	
		support_bundle export	Displays the bundle ID	
		support_bundle set	Displays purge limit for the support bundle and triggers that initiate automatic bundle collection	
8.	Changed	<p>'exporter nodes' are referred to as 'CNG nodes' in the content of the document. The 'cscli export' command is deprecated. Please use 'cscli cng' instead for all --help outputs of 'cscli export', including all its subcommands.</p> <p>NOTE: If the 'cscli export' command is used the following message will be displayed: "This command is deprecated. Please use 'cscli cng' instead."</p>		
9.	Changed/Updated	Advanced Users Commands (Lustre Users Commands)		
10.	Changed	'Customer Installation' is referred to as 'site configuration'		
11.	Deleted	Lustre Users setup from 'Site Configuration' (previously called 'Customer Installation')		
12.	Feature added in rebuild_rate command	The rebuild_rate command displays the numerical values when the rebuild is in progress		
13.	Changed	<ul style="list-style-type: none"> ▪ The cscli cng 'network' command is deprecated. Please use cscli cng 'clientgroup' instead. ▪ The cscli cng share cifs 'network' is deprecated. Please use cscli cng share cifs 'clientgroup' instead. ▪ The cscli cng share nfs 'network' is deprecated. Please use cscli cng share nfs 'clientgroup' instead. 		

7.3 CLI command summary

Table 3 summarizes the CLI commands, with columns indicating the mode or modes that include each command.

Table 3. CLI Command Summary

Site Config	Guest Mode	Daily Mode	Command	Description	Page
<i>Network Setup</i>					61
x	x		show_network_setup	Shows a Sonexion network setup	61
x			set_network	Specifies a Sonexion network setup	61
x			apply_network_setup	Applies a network setup to a Sonexion system	62
x			reset_network_setup	Resets the network setup of an existing Sonexion system	62
<i>Lustre User Commands</i>					63
		x	lustre_users	Configures Lustre filesystem users	63
		x	lustre_users show		64
		x	lustre_users local		65
		x	lustre_users ldap		67
		x	lustre_users ad		70
		x	lustre_users nis		71
		x	lustre_users upcall		73
		x	lustre_users order		74
		x	lustre_users apply		75
<i>System Alert</i>					76
x	x	x	alerts	Displays current and historical system health alerts	76
x		x	alerts_config	Shows and updates the alerts configuration	80
x		x	alerts_notify	Enables or disables alerts notifications	87
<i>Node Control</i>					88
x		x	autodiscovery_mode	Enables or disables auto-discovery mode on system nodes	89
		x	failback	Fails back resources for the specified node	89
		x	failover	Fails over resources to the specified node	89
x		x	mount	Mounts the Lustre file system in the cluster	90
x		x	unmount	Unmounts Lustre clients or targets on the file system	90
		x	power_manage	Specifies node power management options	90

Site Config	Guest Mode	Daily Mode	Command	Description	Page
	x	x	show_nodes	Displays node information	91
<i>Administration</i>					92
x	x	x	fs_info	Retrieves file system information	92
x	x	x	fru	Retrieves FRU (replacement) information	92
x		x	cluster_mode	Toggles the system between 'daily mode' and 'CustWizard' (Site Configuration) mode	93
x	x	x	list	Lists all supported commands	94
x	x	x	syslog	Retrieves syslog entries	94
x		x	batch	Runs a sequence of CSCLI commands in a batch file	95
x		x	set_admin_passwd	Changes administrator user password on an existing Sonexion system	95
x		x	ip_routing	Manages IP routing	95
<i>Configuration</i>					97
		x	configure_hosts	Configures host names for discovered nodes	97
		x	configure_oss	Configures a new OSS node	97
		x	restore_mgmt	Enables MGMT node recovery. When enabled, MGMT node that boots is restored from the latest good backup.	98
		x	show_new_nodes	Displays a table with new OSS nodes and their resources	99
<i>Service Console</i>					99
		x	service_console	Enables or disables the service console or displays the state of the service console	99
<i>Filter Commands</i>					100
		x	create_filter	Creates customer filters for nodes	100
		x	delete_filter	Deletes customer filters for nodes	101
	x	x	show_filters	Shows customized and predefined node filters	100
<i>Update System Software</i>					100
		x	set_node_version	Changes the image for a diskless node	
		x	update_node	Updates the software version on the specified node	101
<i>HA Network Probing - Ibstat</i>					
		x	ibstat_check	Disables/Enables HA's probing of the high speed network	102

Site Config	Guest Mode	Daily Mode	Command	Description	Page
<i>Manage nodes</i>					103
		x	show_version_nodes	Shows all nodes at the specified software version	103
		x	show_node_versions	Shows the current software version on the specified nodes	103
		x	show_update_versions	Shows available software versions in the Sonexion Management Server repository	104
<i>Manage Node Position</i>					103
		x	get_rack_position	Indicates the specified node's position in the Sonexion rack	104
		x	set_rack_position	Changes a given node position in the Sonexion rack	105
<i>Monitor the System's Health</i>					105
x	x	x	monitor	Monitors the current health of the cluster nodes and elements	106
x		x	netfilter_level	Manages the netfilter level	115
<i>RAID Management</i>					115
		x	raid_check	Enables RAID checks on RAID devices	115
x		x	rebuild_rate	Manages the RAID rebuild rate	116
<i>Administrative Settings</i>					117
		x	set_admin_passwd	Manages the administrative password	117
		x	set_date	Manages the date setting on the Sonexion system	118
		x	set_timezone	Manages the timezone setting on the Sonexion system	118
x		x	sm	Manages the InfiniBand Subnet Manager	119
		x	manage_guest	Manages the guest account	119
<i>Support Bundles</i>					120
x	x	x	support_bundle	Manages support bundles and support bundle settings	120

7.4 Network Setup Commands

The `network_setup` command manages network parameters for the Lustre file system. This command includes functions to show, set, apply, and reset Lustre network parameters.

7.4.1 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
<code>-h</code> <code>--help</code>	Displays the help message and exits.
<code>-c</code> <i>cluster_name</i> <code>--cluster</code> <i>cluster_name</i>	Specifies the cluster name.

7.4.2 Set Network Parameters

Modes: Site configuration

The `set_network` command specifies new Lustre network parameters and adds them to the database.

Synopsis

```
$ cscli set_network [-h] -k netmask -r ipranges [-d dns] [-t ntp]  
[-c cluster_name]
```

Optional Arguments	Description
<code>-h</code> <code>--help</code>	Displays the help message and exits.
<code>-k</code> <i>netmask</i> <code>--netmask</code> <i>netmask</i>	Specifies the network mask value of the IP address.
<code>-r</code> <i>ipranges</i> <code>--range</code> <i>ipranges</i>	Specifies the IP address range.
<code>-d</code> <i>dns</i> <code>--dns</code> <i>dns</i>	Specifies the DNS server IP address (optional).

<code>-t ntp --ntp ntp</code>	Specifies the NTP server's IP address (optional).
<code>-c cluster_name --cluster cluster_name</code>	Specifies the cluster name.

7.4.3 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
<code>-h --help</code>	Displays the help message and exits.
<code>-y --yes</code>	Confirms the action to reset the network parameters.
<code>-c cluster_name --cluster cluster_name</code>	Specifies the cluster name.

7.4.4 Apply Network Parameters

Modes: Site configuration

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
<code>-h --help</code>	Displays the help message and exits.
<code> --yes</code>	Confirms the action that network setup parameters were applied.
<code>-c cluster_name --cluster cluster_name</code>	Specifies the cluster name.

7.5 Lustre User Commands

Modes: Daily

The advanced users (Lustre Users) commands support local user and group authentication commands. The following are available for lustre users.

7.5.1 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}
```

Positional Arguments	Description
<code>show</code>	Displays \ detailed information about all Lustre Users settings.
<code>apply</code>	Apply Lustre users configuration.
<code>local</code>	Define and manage local users and groups.
<code>ldap</code>	Management of LDAP settings.
<code>ad</code>	Management of AD settings.
<code>nis</code>	Management of NIS settings.
<code>order</code>	Instruct Lustre upcall to process user/group services in the order specified.
<code>upcall</code>	Configure the underlying method of Lustre upcall.

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.

7.5.2 Subcommand (lustre_users show)

The `lustre_users show` is a subcommand of the `lustre_users` command. This command shows detailed information about all Lustre Users settings.

Synopsis

```
$ cscli lustre_users show [-h]
```

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)
```

7.5.3 Subcommand (lustre_users local)

The `lustre_users local` is a subcommand of the `lustre_users` command. 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
<code>get_all</code>	Downloads Lustre users/groups files.
<code>show</code>	Displays Lustre users/groups settings.
<code>set_all</code>	Uploads Lustre users/groups files.
<code>clear</code>	Resets Lustre users/groups to pristine state.

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.

Subcommand (`lustre_users local get_all`)

The `lustre_users local get_all` is the 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

```
$ csccli lustre_users local get_all [-h] [-u users_file]  
[-g groups_file] [-y]
```

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.
<code>-u <i>users_file</i></code> <code> --users-file <i>users_file</i></code>	The name that is used to save the users file (default <code>lustre-users.txt</code>)
<code>-g <i>groups_file</i> --groups-file <i>groups_file</i></code>	The name that is used to save the groups file (default <code>lustre=groups.txt</code>)
<code>-y --yes</code>	Overwrites an existing file of the same name without prompt.

Subcommand (lustre_users local set_all)

The `lustre_users local set_all` is the 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
<code>-h --help</code>	Displays the help message and exits.
<code>-u <i>users_file</i></code> <code> --users-file <i>users_file</i></code>	A path to a file in SysV '/etc/passwd' format.
<code>-g <i>groups_file</i></code> <code> --groups-file <i>groups_file</i></code>	A path to a file in SysV '/etc/group' format.
<code>-y --yes</code>	Overwrites an existing file of the same name without prompt.

Subcommand (lustre_users local clear)

The `lustre_users local clear` is the 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
<code>-h --help</code>	Displays the help message and exits.
<code>-y --yes</code>	Confirms the reset.

Subcommand (lustre_users local show)

The `lustre_users local show` is the 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
```

7.5.4 Subcommand (lustre_users ldap)

The `lustre_users ldap` is a subcommand of the `lustre_users` command. This command configures users and groups via an external LDAP service.

Synopsis

```
$ cscli lustre_users ldap [-h] {clear,set,show}
```

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.

Subcommand (lustre_users ldap set)

The `lustre_users ldap set` is a subcommand of the `lustre_users ldap` command.

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
<code>-h --help</code>	Displays the help message and exits.
<code>-l <i>ldap_uri</i> --ldap-uri <i>ldap_uri</i></code>	URI, for example: <code>ldap://127.0.0.1:389</code> .
<code>-b <i>base_dn</i> --base-dn <i>base_dn</i></code>	Base Domain Name.
<code>-u <i>user_dn</i> --user-dn <i>user_dn</i></code>	User Domain Name.
<code>-G <i>group_dn</i> --group-dn <i>group_dn</i></code>	Group Domain Name.
<code>-s <i>hosts_dn</i> --hosts-dn <i>hosts_dn</i></code>	Hosts Domain Name.
<code>-i <i>bind_dn</i> --bind-dn <i>bind_dn</i></code>	Bind Domain Name.
<code>-p <i>password</i>, --password <i>password</i></code>	Bind password.
<code>--pvt-key <i>filepath</i></code>	Private key file (pem).
<code>--tls-cert <i>filepath</i></code>	TLS certificate file (crt).
<code>--tls-ca <i>filepath</i></code>	TLS ca certificate file (crt).

Subcommand (lustre_users ldap clear)

The `lustre_users ldap clear` is the 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.

Subcommand (lustre_users ldap show)

The `lustre_users ldap show` is the second level subcommand of the `lustre_users ldap` subcommand.

Synopsis

```
$ csccli lustre_users ldap show [-h]
```

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

7.5.5 Subcommand (lustre_users ad)

The `lustre_users ad` is a subcommand of the `lustre_users` command. This command configures users and groups via an external AD service.

Synopsis

```
$ csccli 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.

Subcommand (lustre_users ad show)

The `lustre_users ad show` is the second level subcommand of the `lustre_users ad` subcommand.

Synopsis

```
$ cscli lustre_users ad show [-h]
```

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.

Subcommand (lustre_users ad set)

The `lustre_users ad set` 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
<code>-h --help</code>	Displays the help message and exits.
<code>-l <i>ldap_uri</i> --ldap-uri <i>ldap_uri</i></code>	URI, for example: <code>ldap://127.0.0.1:389</code> .
<code>-b <i>base_dn</i> --base-dn <i>base_dn</i></code>	Base Domain Name.
<code>-i <i>bind_dn</i> --bind-dn <i>bind_dn</i></code>	Bind Domain Name.
<code>-p <i>password</i>, --password <i>password</i></code>	Bind password.

Subcommand (lustre_users ad clear)

The `lustre_users ad clear` is the 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.

7.5.6 Subcommand (lustre_users nis)

The `lustre_users nis` 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.
clear	Removes NIS configuration.

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

Subcommand (lustre_users nis set)

The `lustre_users nis set` 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 <i>nis_server</i> --nis_server <i>nis_server</i>	Specifies this option 1 to 3 times using IP address or fully-qualified domain name.

<code>-d nis_domain</code> <code> --nis_domain nis_domain</code>	NIS domain. Example: nisdomain.
---	---------------------------------

Subcommand (lustre_users nis clear)

The `lustre_users nis clear` is the 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
<code>-h --help</code>	Displays the help message and exits.
<code>-y --yes</code>	Confirms the configuration is cleared.

Subcommand (lustre_users nis show)

The `lustre_users nis show` is the second level subcommand of the `lustre_users nis` subcommand.

Synopsis

```
$ cscli lustre_users nis show [-h]
```

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.

7.5.7 Lustre Users Upcall Authentication Command

The `lustre_users upcall` command must have a unique User ID. The administrator will receive a warning if the number of UIDs exceed 5000 users. This operation will not fail if the threshold is exceeded. The `lustre_users upcall` command configures the underlying method of Lustre upcall.

Synopsis

```
$ cscli lustre_users upcall [-h] {set,show}
```

Positional Arguments	Description
<code>-show</code>	Displays the underlying method of Lustre upcall.

<code>-set</code>	Configures the underlying method of Lustre upcall.
-------------------	--

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.

Subcommand (lustre_users upcall set)

The `lustre_users upcall set` is a subcommand of the `lustre_users upcall` 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.

NOTE: The Lustre filesystem must be unmounted in order to change upcall type.

Synopsis

```
$ cscli lustre_users upcall set [-h] -u {ad_only,generic}
```

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.
<code>-u {ad_only,generic}</code> <code> --upcall {ad_only,generic}</code>	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.

Subcommand (lustre_users upcall show)

The `lustre_users upcall show` is the second level subcommand of the `lustre_users upcall` subcommand.

This command shows the underlying method of Lustre upcall.

Synopsis

```
$ cscli lustre_users upcall show [-h]
```

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.

7.5.8 Lustre Users Order

The `lustre_users order` command 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/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.

Subcommand (lustre_users order set)

The `lustre_users order set` is a subcommand of the `lustre_users order` 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.

Subcommand (lustre_users order show)

The `lustre_users order show` is the second level subcommand of the `lustre_users order` subcommand.

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.

7.5.9 Lustre Users Apply

The `lustre_users apply` command applies 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.

7.5.10 New and Old Commands Compared

Table 4. New and Old `luster_users` Commands Comparison

New Commands	Old Commands
<code>lustre_users</code>	None
<code>lustre_users show</code>	<code>get_lustre_users_ldap</code> <code>get_lustre_users_nis</code> <code>get_lustre_users_upcall</code>
<code>lustre_users local</code>	None
<code>lustre_users ldap set</code>	<code>set_lustre_users_ldap</code>
<code>lustre_users ldap clear</code>	<code>set_lustre_users_ldap -C</code>
<code>lustre_users nis set</code>	<code>set_lustre_users_nis</code>

New Commands	Old Commands
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

7.6 System Alert Commands

Alert commands include functions to view and update the alerts configuration, turn on/off alert notifications, and display current and historical system alerts.

7.6.1 Displaying Current and Historic System Alerts

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.
thresholds	Displays editable alert thresholds and their current settings.

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

Subcommand (alerts elements_active)

Synopsis

```
$ csccli 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 <i>node_spec</i> --nodes <i>node_spec</i>	Specifies pdsh-style node hostnames (e.g. node[100-110,120]).
-g <i>genders_query</i> --genders <i>genders_query</i>	Specifies node genders attributes query (e.g. mds=primary).
-S <i>element_filter</i> --search <i>element_filter</i>	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.

Subcommand (alerts nodes)

Synopsis

```
$ csccli 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 <i>start_time</i> --start-time <i>start_time</i>	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.
-e <i>end_time</i> --end-time <i>end_time</i>	Specifies the alert end time in ISO-8601 format. (Default value is "now".)

-m <i>limit</i> --max <i>limit</i>	Specifies the maximum number (limit) of alerts to display.
-n <i>node_name</i> --node <i>node_name</i>	Specifies the node for which to display alerts. Pdsh-style node masks are <u>not</u> allowed here.
-N {down,unreachable,up} --node status	Specifies node status.

Subcommand (alerts elements)

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 <i>start_time</i> --start-time <i>start_time</i>	Specifies the start time filter in ISO-8601 format. If --start-time is not specified, then --end-time is ignored and the "last 7 days" period is used.
-e <i>end_time</i> --end-time <i>end_time</i>	Specifies the end time filter in ISO-8601 format (default value is "now").
-m <i>limit</i> --max <i>limit</i>	Specifies the maximum number (limit) of items to display.
-n <i>node_name</i> --node <i>node_name</i>	Specifies the node for which to display items. Pdsh-style node masks are <u>not</u> allowed here.
-U {unknown,warning,ok,critical} --element status	Specifies the element's status.

Subcommand (alerts nodes_active)

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.

<code>-y --yaml</code>	Outputs data in YAML format.
<code>-v --verbose</code>	Outputs extra data.
<code>-x --unhandled</code>	Displays alerts for notifications that have not been turned off (default is all alerts are shown).
<code>-n <i>node_spec</i> --nodes <i>node_spec</i></code>	Specifies pdsh-style node hostnames (e.g. <code>node[100-110,120]</code>).
<code>-g <i>genders_query</i> --genders <i>genders_query</i></code>	Specifies node genders attributes query (e.g. <code>mds=primary</code>).

Subcommand (alerts threshold)

Synopsis

```
$ cscli alerts thresholds [-h] [-y]
```

Threshold fields:

<code>name</code>	Short identifier of the threshold
<code>description</code>	Describes the threshold and gives tips on how to modify it
<code>gender</code>	Type of nodes to which the threshold is applied
<code>warning</code>	Value of the warning threshold
<code>critical</code>	Value of the critical threshold

Possible gender values:

<code>all</code>	All nodes; general node type that can be overwritten by more specific node types
<code>mgmt</code>	Management nodes (primary and secondary)
<code>mds</code>	Metadata Servers
<code>oss</code>	Object Storage Servers

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.
<code>-y --yaml</code>	Outputs data in YAML format.

7.6.2 Manage the Alerts Configuration

Modes: Site configuration, Daily

The `alerts_config` command enables you to view and update 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
<code>email_off</code>	Turns off notifications for notification subscribers.
<code>thresholds</code>	Sets the current value of an threshold. This value can be edited
<code>email_update</code>	Sends an email alert with an update.
<code>email_server_update</code>	Sends an email alert with a server update.
<code>email_delete</code>	Deletes the email.
<code>email_add</code>	Adds a new notification subscriber.
<code>email_on</code>	Turns on notifications for subscribers.
<code>email_server</code>	Displays the relay SMTP server configuration.
<code>emails</code>	Lists the alert notification subscribers.

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.

Subcommand (`email_off`)

The `email_off` command turns off notifications for subscribers.

Synopsis

```
$ cscli alerts_config email_off [-h] -u email
```

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.

<code>-u email --user email</code>	Displays subscriber email. Notifies subscribers have new mail in /var/spool/mail or admin.
--------------------------------------	---

Subcommand (thresholds)

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
<code>-h --help</code>	Displays the help message and exits.
<code>-t <i>threshold_name</i></code> <code> --threshold <i>threshold_name</i></code>	Displays the name of the threshold.
<code>-g <i>gender_name</i></code> <code> --gender <i>gender_name</i></code>	Displays the gender name of the threshold.
<code>-W <i>warning_threshold_value</i></code> <code> --warning <i>warning_threshold_value</i></code>	Displays the warning threshold value.
<code>-C <i>critical_threshold_value</i></code> <code> --critical <i>critical_threshold_value</i></code>	Displays the critical threshold value.
<code>-A --apply-config</code>	Applies the threshold configuration.

Note, 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.

If you would like to make a group of changes to the thresholds, you may 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.

Subcommand (email_update)

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

Notification Periods

The Notification period are:

- 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.

<code>-u email --user email</code>	Displays subscriber email. Notifies you have new mail in /var/spool/mail or admin.
<code>-M email --email email</code>	Displays the email address.
<code>-N user_full_name --name user_full_name</code>	Displays a longer name or description for the subscriber.
<code>-P {24x7,workhours} --period {24x7,workhours}</code>	Displays the time periods at which the subscriber is notified. possible values: {24x7,workhours}
<code>-L level, --level level</code>	Displays notification level; possible values: any comma-separated combination of {critical,ok,unknown,warning}, or "all", or "none".

Subcommand (email_server_update)

The `email_server_update` command configures the SMTP server to send alerts to external email addresses.

Synopsis

```
$ csccli 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
<code>-h --help</code>	Displays the help message and exits.
<code>-s smtp_server_address --server smtp_server_address</code>	Displays an IP address or hostname of the (relay) SMTP server.
<code>--port port</code>	SMTP server port (default: 25)
<code>-S email_from --sender email_from</code>	Displays the senders email address. If the <code>--domain</code> is set, the default value for the sender is <code>cluster_name@domain</code> . If the <code>--domain</code> is not set, the <code>--sender</code> 's emails address is required.

<code>-d domain</code> <code> --domain domain</code>	Displays the internet hostname of the mail system to be used with email addresses that have no "@".
<code>-u smtp_user, --user smtp_user</code>	Specifies the username if the SMTP server requires authentication.
<code>-p smtp_password</code> <code> --password smtp_password</code>	The password if the SMTP server requires authentication.

Subcommand (email_delete)

The `email_delete` deletes notifications to subscribers.

Synopsis

```
$ cscli alerts_config email_delete [-h] -u email
```

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.
<code>-u email --user email</code>	Displays subscriber email. Notifies you have new mail in <code>/var/spool/mail</code> or <code>admin</code> .

Subcommand (email_add)

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:

The possible Levels 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 <i>email</i> --email <i>email</i>	Displays subscriber email. Notifies you have new mail in /var/spool/mail or admin. email address.
-N <i>user_full_name</i> --name <i>user_full_name</i>	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 <i>level</i> --level <i>level</i>	The notification level. Possible values: any comma-separated combination of: {critical,ok,unknown,warning}, or "all", or "none" (default: all).

Subcommand (email_on)

The `email_on` command turns on notifications for subscribers.

Synopsis

```
$ csccli alerts_config email_on [-h] -u email
```

Optional Arguments	Description
-h --help	Displays the help message and exits.
-u <i>email</i> --user <i>email</i>	Displays subscriber email. Notifies you have new mail in /var/spool/mail or admin.

Subcommand (email_server)

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.

Subcommand (emails)

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:

The possible Levels 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.

<code>-u email --user email</code>	Displays subscriber email. Notifies you have new mail in /var/spool/mail or admin.
--------------------------------------	---

7.6.3 Managing the Alerts Notification

Modes: Site configuration, Daily

The `alerts_notify` command turns alert notifications on or off.

Synopsis

```
$ csccli alerts_notify [-h] {on,off} ...
```

Positional Arguments	Description
<code>on</code>	Sets the alert notification on.
<code>off</code>	Sets the alert notification off.

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.

`alerts_notify on`

The `alerts_notify on` command turns alert notifications on.

Synopsis

```
$ csccli alerts_notify on [-h] (-n node_spec | -g genders_query)
  [-S element_filter | -E element_name]
```

Positional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.
<code>-n node_spec --node --node_spec --nodes node_spec</code>	Looks through passed hostname elements. Looks for pdsh style nodes host names (e.g. node[100-110,120]).
<code>-g genders_query --genders genders_query</code>	Displays the node genders attributes query (e.g. mds=primary).
<code>-S element_filter --search element_filter</code>	This command searches by element name. The pattern is case sensitive. Regular expressions allowed.

-E <i>element_name</i> --element <i>element_name</i>	Displays the element name.
--	----------------------------

alerts_notify on

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 <i>node_spec</i> --node -- <i>node_spec</i> --nodes <i>node_spec</i>	Looks through passed hostname elements. Looks for pdsh style nodes host names (e.g. node[100-110,120]).
-g <i>genders_query</i> --genders <i>genders_query</i>	Displays the node genders attributes query (e.g. mds=primary).
-S <i>element_filter</i> --search <i>element_filter</i>	This command searches by element name. The pattern is case sensitive. Regular expressions allowed.
-E <i>element_name</i> --element <i>element_name</i>	Displays element name.
-C <i>comment</i> --comment <i>comment</i>	Displays a brief description of what you are doing.

7.7 Node Control Commands

The node control commands are used to control individual Lustre nodes (MDS/MGS and OSSs) 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, managing node auto-discovery and controlling CNG nodes.

7.7.1 Managing Node Auto-Discovery

Modes: Site configuration, Daily

This 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.

7.7.2 Managing Node Failback and Failover

Modes: Daily

These 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 <i>filter_sid</i> --filter <i>filter_sid</i>	The filter identifier for the specified node. Failover/failback actions run on the nodes by filtering this filter.
-n <i>node_spec</i> --nodes <i>node_spec</i>	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 <i>cluster_name</i> --cluster <i>cluster_name</i>	This parameter is deprecated. It is supported only for backward compatibility.

7.7.3 Mounting and Unmounting Lustre Targets

Modes: Site configuration, Daily

The `mount` and `unmount` commands control file system access to the Lustre targets (MDS/MGS and OSSs). 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.

IMPORTANT: 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
<code>-h</code> <code>--help</code>	Displays the help message and exits.
<code>-f</code> <i>fs_name</i> <code>--fs-name=</code> <i>fs_name</i>	Specifies the name of the file system.
<code>-n</code> <i>node_spec</i> <code>--nodes=</code> <i>node_spec</i>	Specifies the node(s) on which the mount/unmount action is performed. Node hostnames should be passed in pdsh style.
<code>-c</code> <i>cluster_name</i> <code>--cluster</code> <i>cluster_name</i>	This parameter is deprecated. It is supported only for backward compatibility.
<code>--evict</code>	This parameter evicts clients before they unmount.
<code>--force</code>	This parameter is the force mode to evict lustre clients.

7.7.4 Managing Node Power

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 <i>filter_sid</i> --filter <i>filter_sid</i>	The filter identifier for the specified node. Failover/failback actions run on the nodes by filtering this filter. If --filter is specified, then --nodes is ignored.
-n <i>node_spec</i>	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.
--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 <i>force</i> mode; should only be used with --power-off.
-c <i>cluster_name</i> --cluster <i>cluster_name</i>	This parameter is deprecated. It is supported only for backward compatibility.

7.7.5 Showing Node Information

Modes: Guest, Daily

This command displays information about specified system nodes.

Synopsis

```
$ cscli show_nodes [-h] [-F filter_sid] [-r] [-c cluster_name]
    | --cluster cluster_name
```

Option	Description
-h --help	Displays the help message and exits.
-F <i>filter_sid</i> --filter <i>filter_sid</i>	Specifies the node filter.
-r --refresh	Specifies the refresh mode (press 'q' for quit).
-c <i>cluster_name</i> --cluster <i>cluster_name</i>	This parameter is deprecated. It is supported only for backward compatibility.

7.8 Administrative Commands

Administrative commands include functions to get file system and cluster node information, retrieve syslog entries, show FRU information and list available commands.

7.8.1 Show File System Information

Modes: Site configuration, Guest, Daily

The `fs_info` command shows all file system information.

Synopsis

```
$ cscli fs_info [-h] [-f fs_name] [-c cluster_name] | --cluster cluster_name
```

Optional Arguments	Description
-h --help	Displays the help message and exits.
-f <i>fs_name</i> --fs <i>fs_name</i>	Displays the file system name.
-c <i>cluster_name</i> --cluster <i>cluster_name</i>	This parameter is deprecated. It is supported only for backward compatibility.

7.8.2 Retrieving FRU Information

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

```
$ csccli fru [-h] (-a | -n node_spec) [-t
    {ArrayDevice,BMC,Cooling,Enclosure,Enclosure_Electronics,PSU,Battery}]
    [-i index] [-l [history]]
```

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 <i>node_spec</i> --nodes <i>node_spec</i>	Displays FRUs (including status) grouped by element type, for a specified node(s) in the system.
-t {ArrayDevice, <i>bmc</i> ,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 <i>index</i> --index <i>index</i>	Displays FRUs (including status) for specified elements within a list of elements of the same type.
-l [<i>history</i>] --history [<i>history</i>]	Displays FRU history (default is 10 lines of history).

7.8.3 Change the Sonexion Mode

Modes: Site configuration, Daily

The `cluster_mode` command toggles the Sonexion system among multiple system modes: daily, custWizard or pre-shipment.

Synopsis

```
$ csccli 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 either daily mode, site configuration mode or pre-shipment mode.

<code>--db-only</code>	Update only the database. Does not sync nodes via puppet. Valid only with ' <code>--mode</code> ' argument.
------------------------	---

7.8.4 Listing Commands

Modes: Site configuration, Guest, Daily

The `list` command shows a list of available commands in the current Sonexion mode.

Synopsis

```
$ cscli list [-h]
```

Optional Arguments	Description
<code>-h</code> <code>--help</code>	Displays the help message and exits.

7.8.5 Displaying Log Information

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
<code>-h</code> <code>--help</code>	Displays the help message and exits.
<code>-m <i>max</i></code> <code>--max=<i>max</i></code>	Specifies the maximum number of entries to return.
<code>-F</code> <code>--follow</code>	Polls for future messages. Only valid without <code>-e</code> , <code>-r</code> arguments.
<code>-d <i>duration</i></code> <code>--duration=<i>duration</i></code>	Specifies duration (in seconds) for which to follow output. Only valid with <code>-F</code> argument.
<code>-s <i>start_time</i></code> <code>--start_time=<i>start_time</i></code>	Specifies the earliest time for which messages should be received.
<code>-e <i>end_time</i></code> <code>--end_time <i>end_time</i></code>	Specifies the latest time for which messages should be received.
<code>-r</code> <code>--reverse</code>	Sorts entries in descending order (by time).

7.8.6 Setting Administrator Password

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
<code>-h --help</code>	Displays the help message and exits.
<code>-p --password</code>	Specify the new administrator password string.

7.8.7 Batching Commands

Modes: Site configuration, Daily

The `batch` command runs a sequence of CSCLI commands in a batch file.

Synopsis

```
$ cscli batch [-h] -b batch_file
```

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.
<code>-b <i>batch_file</i></code> <code> --batch-file <i>batch_file</i></code>	Specifies the command batch file.

7.8.8 Managing IP Routing

Modes: Site configuration, Daily

The `ip_routing` command manages IP routing to and from the system database.

Synopsis

```
$ cscli ip_routing [arguments]
```

where [arguments] are:

```
--show|-s [--loadable]
```

or

```
--load path_to_file
```

or

```

--add | -a --dest destination_ip --prefix prefix_len
--router router_ip

OR

--update | -u --route-id route_id [--destdestination_ip]
[--prefixprefix_len] [--routerrouter_ip]

OR

--delete | -d --route-id route_id

OR

--clear | -c

OR

--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 <code>-show</code> argument).
-c --clear	Clears the routing table in the database.
--apply	Applies IP routing.
--load <i>load</i>	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 <i>dest</i>	Specifies the destination IP address.
--prefix <i>prefix</i>	Specifies the prefix length (0-32).
--router <i>router</i>	Specifies the router IP address.
--route-id <i>route_id</i>	Specifies the route identifier (see <code>ip_routing -show</code>).

7.9 Configuration Commands

The configuration commands specifies the MAC address and hostname for a given node and configures OSS nodes

7.9.1 Configuring Hosts

Modes: Daily

The `configure_hosts` command configures the MAC address and host names for the discovered node.

Synopsis

```
$ cscli configure_hosts [-h] -m mac_address --hostname hostname [-f]
```

Optional Arguments	Description
<code>-h</code> <code>--help</code>	Displays the help message and exits.
<code>-m</code> <i>mac_address</i> <code>--mac</code> <i>mac_address</i>	Displays the <i>mac_address</i> and node MAC address.
<code>--hostname</code> <i>hostname</i>	Displays the new node hostname.
<code>-f</code> <code>--force</code>	Forces the mode (to skip hostname validation).

7.9.2 Configuring New OSS Nodes

Modes: Daily

This 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
<code>-h</code> <code>--help</code>	Displays the help message and exits.
<code>-n</code> <i>node_spec</i> <code>--nodes</code> <i>node_spec</i>	Specifies the hostname of the new OSS node (in genders style).
<code>-A</code> <code>--apply-config</code>	Applies the configuration to the new OSS node.

-b <i>bind_arrays</i> --bind-arrays <i>bind_arrays</i>	Specifies comma-separated pairs of array-file system bindings. Each binding should be in this format: <i>array:file_system_name</i> . The <i>array</i> variable can be a genders-style string, e.g. md[0-3].
-c <i>cluster_name</i> --cluster <i>cluster_name</i>	This parameter is deprecated. It is supported only for backward compatibility.

7.9.3 Restoring Management Nodes

Modes: Daily

This command enables MGMT node recovery. When enabled, the MGMT node that boots is restored from the latest good backed up nodes. The `restore_mgmt` 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 will copy 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 `--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.

7.9.4 Showing Information About New OSS Nodes

Modes: Daily

This 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 <i>cluster_name</i> --cluster <i>cluster_name</i>	This parameter is deprecated. It is supported only for backward compatibility.

7.10 Service Commands

In this release a service console command is added to service components.

The purpose of the `service_console` command is to allow a site administrator to block or allow a service technician from using the system service console. 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 which 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.

Synopsis

```
$ cscli service_console [-h] (-e | -d | -s)
```

Optional Arguments	Description
-h --help	Displays the help message and exits.
-e --enable	Enables the service console.
-d --disable	Disables the service console.

<code>-s --show</code>	Displays the state if the service console.
-------------------------	--

7.11 Filter Commands

The filter commands create and delete a filter.

7.11.1 Creating 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
<code>-h --help</code>	Displays the help message and exits.
<code>-i <i>filter_sid</i></code> <code> --id <i>filter_sid</i></code>	Displays the symbol identifier of the filter.
<code>-F <i>filter_name</i></code> <code> --name <i>filter_name</i></code>	Displays the filter name.
<code>-e <i>filter_expr</i></code> <code> --expression <i>filter_expr</i></code>	Displays the filter expression. Examples: "host1,host2", "host[1-3]", "mds=primary".

7.11.2 Showing Filters

Modes: Guest, Daily

The `show_filters` command shows all filters.

Synopsis

```
$ cscli show_filters [-h] [-P] [-C]
```

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.
<code>-P --predefined</code>	Displays only predefined filters.

<code>-C --custom</code>	Displays only custom filters.
---------------------------	-------------------------------

7.11.3 Deleting a Filter

Modes: Daily

The `delete_filter` command deletes a customer nodes filter.

Synopsis

```
$ cscli delete_filter [-h] -i filter_sid
```

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.
<code>-i <i>filter_sid</i></code> <code> --id <i>filter_sid</i></code>	Displays the symbol identifier of the filter.

7.12 Update System Software

These commands prepare a software upgrade package for installation and apply it to system nodes.

7.12.1 Update Software on a System Node

Modes: Daily

The `update_node` command updates software on the specified node(s).

Synopsis

```
$ cscli update_node [-h] -n node_spec
```

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.
<code>-n <i>node_spec</i></code> <code> --node-spec <i>node_spec</i></code>	Specifies hostnames of the nodes on which to update software.

7.12.2 Set a Node Version

Modes: Daily

The `set_node_version` command is used to change the image of diskless node(s). It's 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 "admin" account only.

Synopsis

```
$ cscli set_node_version [-h] -n nodes -v version
```

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.
<code>-n nodes --node nodes</code>	Displays pdsh style nodes hostnames.
<code>-v version --version version</code>	Version of the image to use.

7.13 Iostat Check Command

The `ibstat_check` is part of HA stack settings. It disables and enables HA's probing of high speed network(s). This command is available for "admin" account only.

Synopsis

```
$ cscli ibstat_check [-h] [-n nodes] (--enable | --disable)
```

Option	Description
<code>-h --help</code>	Displays the help message and exits.
<code>-n nodes --node nodes</code>	Displays pdsh-style nodes hostnames. Note: Lustre Server Nodes only.
<code>--enable</code>	Enables HA's probing of the high speed network.
<code>--disable</code>	Disables HA's probing of the high speed network.

7.14 Manage Nodes

7.14.1 Show Nodes at Specified Software Version

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
<code>-h --help</code>	Displays the help message and exits.
<code>-q --query</code>	Controls the format of the command output. If this flag is specified, nodes in output should display in genders style, e.g. <code>mycluster[02-05,97-98]</code> .
<code>-v <i>sw_version</i></code> <code> --version <i>sw_version</i></code>	Specifies the Sonexion software version.

7.14.2 Show Node Versions

Modes: Daily

This 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
<code>-h --help</code>	Displays the help message and exits.
<code>-q, --query</code>	Controls output format. If this flag is specified, nodes in the output should be in genders style. For example, <code>(mycluster[02-05,97-98])</code>
<code>-n <i>node_spec</i>,</code> <code>--nodes <i>node_spec</i></code>	Specifies nodes to indicate the Sonexion software version.
<code>-g <i>genders_query</i></code>	Specifies a gender's style query.

7.14.3 Show Available Software Versions

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.

7.15 Manage Node Position in a Rack

The rack position commands manage the location of components (hosting system nodes) in a Sonexion rack. The CMU hosts the primary and secondary MGMT, MGS and MDS nodes. Each SSU hosts OSS nodes (two OSSs per SSU).

7.15.1 Get Node Position in a Sonexion Rack

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 <i>racks</i> --racks <i>racks</i>	Displays the hostnames in all positions within the specified racks. <i>racks</i> is a comma-separated list of rack names.
-a --all	Displays information about all racks.

7.15.2 Setting Node Position in a Sonexion Rack

Modes: Daily

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 <i>yaml_path</i> --yaml <i>yaml_path</i>	Loads rack position information from <i>yaml</i> file format.
-r <i>rack_name</i> --rack <i>rack_name</i>	Specifies the rack containing the node(s). This can be set manually.
-n <i>node_spec</i> --node <i>node_spec</i>	Specifies the node(s) hostname. This can be set manually.
-p <i>position</i> --position= <i>position</i>	Specifies the node position in rack units. This can be set manually.

7.16 Monitor the System's Health

Modes: Site configuration, Guest, Daily

The `$ cscli 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 - status summary.

<code>nodes</code>	Current status for nodes.
<code>elements</code>	Current status for elements.

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.

Subcommand (monitor nodes)

The `$ cscli monitor nodes` command monitors individual nodes.

Synopsis

```
$ cscli monitor nodes [-h] [-y] [-v] [-n node_spec | -g genders_query]
  [-N {down,unreachable,up,pending}]
```

Positional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.
<code>-y --yaml</code>	Displays output data in YAML format.
<code>-v --verbose</code>	Outputs extra data.
<code>-n <i>node_spec</i> --node <i>node_spec</i></code> <code> --nodes <i>node_spec</i></code>	Looks through passed hostname elements. Looks for pdsh style nodes host names (e.g. <code>node[100-110,120]</code>).
<code>-g <i>genders_query</i></code>	Displays the node genders attributes query (e.g. <code>mds=primary</code>).
<code>-N {down,unreachable,up,pending}</code> <code> --nodestatus</code> <code>{down,unreachable,up,pending}</code> <code>node status.</code>	Displays node status.

Subcommand (monitor elements)

The `$ cscli 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.
-n <i>node_spec</i> --node <i>node_spec</i> --nodes <i>node_spec</i>	Looks through passed hostname elements. Looks for pdsh style nodes host names (e.g. node[100-110,120]).
-g <i>genders_query</i>	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 <i>element_filter</i> --search <i>element_filter</i>	Searches by element name. The pattern is case sensitive. Regular expressions are allowed.

NOTE: If you call this command without any options, you may get thousands of elements on a large system.

CSCLI Monitor command examples and outputs are given below including OK, WARNING and CRITICAL.

Examples

```
[root@snx11000n000 ~]# 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@snx11000n000 ~]# cscli monitor elements -U unknown -U pending -U warning -U
critical
[root@snx11000n000 ~]# cscli monitor elements -U unknown
[root@snx11000n000 ~]# cscli monitor elements -U pending
[root@snx11000n000 ~]# cscli monitor elements -U critical
[root@snx11000n000 ~]# cscli monitor elements -U warning
[root@snx11000n000 ~]# cscli monitor nodes
```

Sonexion® Administrator's Guide

```
snx11000n000: UP for 28d 15h 33m 26s checked 2014-02-06 10:00:36 "PING OK - Packet
loss = 0%, RTA = 0.03 ms"
snx11000n001: UP for 15d 18h 14m 44s checked 2014-02-06 10:02:56 "PING OK - Packet
loss = 0%, RTA = 0.16 ms"
snx11000n002: UP for 15d 18h 6m 54s checked 2014-02-06 10:05:36 "PING OK - Packet
loss = 0%, RTA = 0.18 ms"
snx11000n003: UP for 15d 18h 8m 54s checked 2014-02-06 10:03:36 "PING OK - Packet
loss = 0%, RTA = 0.18 ms"
snx11000n003-Enclosure-R1C1-21U: UP for 28d 15h 35m 8s checked 2014-02-06 09:55:36
"OK"
snx11000n004: UP for 5d 17h 14m 44s checked 2014-02-06 10:01:26 "PING OK - Packet
loss = 0%, RTA = 0.16 ms"
snx11000n005: UP for 5d 18h 30m 14s checked 2014-02-06 10:03:26 "PING OK - Packet
loss = 0%, RTA = 0.19 ms"
snx11000n005-Enclosure-R1C1-5U: UP for 28d 15h 34m 12s checked 2014-02-06 10:02:36
"OK"
[root@snx11000n000 ~]# cscli monitor elements
```

Subset of output:

```
snx11000n000 "Arrays and Disk Status": OK for 28d 15h 33m 49s checked 2014-02-06
10:00:52 "All arrays are operating normally"
snx11000n000 "Current Load": OK for 28d 15h 33m 49s checked 2014-02-06 10:00:52 "OK
- load average: 0.08, 0.03, 0.02"
snx11000n000 "Current Users": OK for 28d 15h 33m 49s checked 2014-02-06 10:00:52
"USERS OK - 1 users currently logged in"
snx11000n000 "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%):"
snx11000n000 "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)"
snx11000n000 "RAM usage": OK for 28d 15h 33m 49s checked 2014-02-06 10:00:52 "OK -
11.6% (3807704 kB) used."
snx11000n000 "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)"
snx11000n000 "Total Processes": OK for 28d 15h 33m 49s checked 2014-02-06 10:00:52
"PROCS OK: 407 processes with STATE = RSZDT"
snx11000n000 "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"
snx11000n000 "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"
snx11000n000 "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"
snx11000n000 "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"
snx11000n000 "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"
snx11000n000 "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"
```

```

snx11000n004 "Arrays and Disk Status": OK for 2d 14h 28m 10s checked 2014-02-06
10:03:42 "All arrays are operating normally"
snx11000n004 "Current Load": OK for 28d 15h 32m 14s checked 2014-02-06 10:03:42 "OK
- load average: 0.01, 0.01, 0.01"
snx11000n004 "Current Users": OK for 28d 15h 32m 14s checked 2014-02-06 10:03:42
"USERS OK - 0 users currently logged in"
snx11000n004 "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%):"
snx11000n004 "Lustre Health": OK for 28d 15h 32m 14s checked 2014-02-06 10:03:42
"OK:Lustre is ok"
snx11000n004 "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)"
snx11000n004 "RAM usage": OK for 28d 15h 32m 14s checked 2014-02-06 10:03:42 "OK -
12.9% (4203984 kB) used."
snx11000n004 "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)"
snx11000n004 "Total Processes": OK for 28d 15h 32m 14s checked 2014-02-06 10:03:42
"PROCS OK: 1239 processes with STATE = RSZDT"
root@snx11000n000 ~]# cscli monitor elements -v

```

Subset of output:

```

snx11000n000 "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: 111101000000
slot: 21, wwn: 5000c500479061af, cap: 450098159616, dev: sdv, parts:
0, status: Hot Spare, t10: 111101000000
MD RAID to Lustre mapping
Array /dev/md/snx11000n003:md64 doesn't have associated WIB array
Degraded Array information:
All arrays are in clean state on node snx11000n000"
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
-----
snx11000n000 "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
-----

```

Sonexion® Administrator's Guide

```
snx11000n000 "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@snx11000n000 ~]# cscli monitor elements -S enclosures
```

```
snx11000n003-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"
snx11000n003-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"
snx11000n003-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"
snx11000n003-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"
snx11000n003-Enclosure-R1C1-21U "Power Statistics": OK for 15d 17h 19m 4s checked
  2014-02-06 09:58:06 "Summary: Total System Power 168W"
snx11000n003-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"
snx11000n003-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"
snx11000n005-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"
snx11000n005-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"
snx11000n005-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"
snx11000n005-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"
snx11000n005-Enclosure-R1C1-5U "Power Statistics": OK for 28d 15h 28m 40s checked
  2014-02-06 09:57:42 "Summary: Total System Power 1068W"
snx11000n005-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"
snx11000n005-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@snx11000n000 ~]#
```

```
[root@snx11000n000 ~]# cscli monitor nodes -n snx11000n004
```

```
snx11000n004: UP for 5d 17h 17m 26s checked 2014-02-06 10:06:36 "PING OK - Packet
  loss = 0%, RTA = 0.17 ms"
```

```
[root@snx11000n000 ~]# cscli monitor elements -n snx11000n004
```

```
snx11000n004 "Arrays and Disk Status": OK for 2d 14h 30m 42s checked 2014-02-06
  10:03:42 "All arrays are operating normally"
snx11000n004 "Current Load": OK for 28d 15h 34m 46s checked 2014-02-06 10:03:42 "OK
  - load average: 0.01, 0.01, 0.01"
```

```

snx11000n004 "Current Users": OK for 28d 15h 34m 46s checked 2014-02-06 10:03:42
  "USERS OK - 0 users currently logged in"
snx11000n004 "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%):"
snx11000n004 "Lustre Health": OK for 28d 15h 34m 46s checked 2014-02-06 10:03:42
  "OK:Lustre is ok"
snx11000n004 "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)"
snx11000n004 "RAM usage": OK for 28d 15h 34m 46s checked 2014-02-06 10:03:42 "OK -
  12.9% (4203984 kB) used."
snx11000n004 "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)"
snx11000n004 "Total Processes": OK for 28d 15h 34m 46s checked 2014-02-06 10:03:42
  "PROCS OK: 1239 processes with STATE = RSZDT"
[root@snx11000n000 ~]# csccli monitor elements -g oss
snx11000n004 "Arrays and Disk Status": OK for 2d 14h 31m 42s checked 2014-02-06
  10:08:43 "All arrays are operating normally"
snx11000n004 "Current Load": OK for 28d 15h 35m 46s checked 2014-02-06 10:08:43 "OK
  - load average: 0.01, 0.01, 0.01"
snx11000n004 "Current Users": OK for 28d 15h 35m 46s checked 2014-02-06 10:08:43
  "USERS OK - 0 users currently logged in"
snx11000n004 "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%):"
snx11000n004 "Lustre Health": OK for 28d 15h 35m 46s checked 2014-02-06 10:08:43
  "OK:Lustre is ok"
snx11000n004 "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)"
snx11000n004 "RAM usage": OK for 28d 15h 35m 46s checked 2014-02-06 10:08:43 "OK -
  12.9% (4204568 kB) used."
snx11000n004 "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)"
snx11000n004 "Total Processes": OK for 28d 15h 35m 46s checked 2014-02-06 10:08:43
  "PROCS OK: 1239 processes with STATE = RSZDT"
snx11000n005 "Arrays and Disk Status": OK for 17d 14h 22m 41s checked 2014-02-06
  10:07:41 "All arrays are operating normally"
snx11000n005 "Current Load": OK for 28d 15h 38m 8s checked 2014-02-06 10:07:41 "OK
  - load average: 0.01, 0.02, 0.02"
snx11000n005 "Current Users": OK for 28d 15h 38m 8s checked 2014-02-06 10:07:41
  "USERS OK - 0 users currently logged in"
snx11000n005 "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%):"
snx11000n005 "Lustre Health": OK for 28d 15h 38m 7s checked 2014-02-06 10:07:42
  "OK:Lustre is ok"
snx11000n005 "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)"
snx11000n005 "RAM usage": OK for 28d 15h 38m 8s checked 2014-02-06 10:07:41 "OK -
  12.8% (4192544 kB) used."
snx11000n005 "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)"
snx11000n005 "Total Processes": OK for 28d 15h 38m 8s checked 2014-02-06 10:07:41
  "PROCS OK: 1241 processes with STATE = RSZDT"
[root@snx11000n000 ~]#

```

```
[root@snx11000n000 ~]# cscli monitor elements -S arrays
snx11000n000 "Arrays and Disk Status": OK for 28d 15h 38m 14s checked 2014-02-06
10:05:52 "All arrays are operating normally"
snx11000n001 "Arrays and Disk Status": OK for 28d 15h 39m 56s checked 2014-02-06
10:08:07 "All arrays are operating normally"
snx11000n002 "Arrays and Disk Status": OK for 28d 15h 36m 38s checked 2014-02-06
10:07:56 "All arrays are operating normally"
snx11000n003 "Arrays and Disk Status": OK for 28d 15h 36m 36s checked 2014-02-06
10:06:24 "All arrays are operating normally"
snx11000n004 "Arrays and Disk Status": OK for 2d 14h 32m 35s checked 2014-02-06
10:08:43 "All arrays are operating normally"
snx11000n005 "Arrays and Disk Status": OK for 17d 14h 23m 34s checked 2014-02-06
10:07:41 "All arrays are operating normally"
[root@snx11000n000 ~]# cscli monitor elements -S arrays -v
```

Subset of output

```
snx11000n000 "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/snx11000n003:md64 doesn't have associated WIB array
Degraded Array information:
All arrays are in clean state on node snx11000n000"
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
-----
snx11000n001 "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/snx11000n003:md67 doesn't have associated WIB array
Degraded Array information:
All arrays are in clean state on node snx11000n001"
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

```
-----
snx11000n002 "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/snx11000n003:md65 doesn't have associated WIB array
Target: MGS
Degraded Array information:
All arrays are in clean state on node snx11000n002"
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@snx11000n000 ~]# cscli monitor elements -S disk
snx11000n000 "Arrays and Disk Status": OK for 28d 15h 43m 32s checked 2014-02-06
  10:10:52 "All arrays are operating normally"
snx11000n001 "Arrays and Disk Status": OK for 28d 15h 45m 14s checked 2014-02-06
  10:13:07 "All arrays are operating normally"
snx11000n002 "Arrays and Disk Status": OK for 28d 15h 41m 56s checked 2014-02-06
  10:12:56 "All arrays are operating normally"
snx11000n003 "Arrays and Disk Status": OK for 28d 15h 41m 54s checked 2014-02-06
  10:11:24 "All arrays are operating normally"
snx11000n004 "Arrays and Disk Status": OK for 2d 14h 37m 53s checked 2014-02-06
  10:13:42 "All arrays are operating normally"
snx11000n005 "Arrays and Disk Status": OK for 17d 14h 28m 52s checked 2014-02-06
  10:12:41 "All arrays are operating normally"
root@snx11000n000 ~]# cscli monitor elements -S fan
snx11000n003-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"
snx11000n003-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"
snx11000n005-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"
snx11000n005-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@snx11000n000 ~]# cscli monitor elements -S power
snx11000n003-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"
snx11000n003-Enclosure-R1C1-21U "Power Statistics": OK for 15d 17h 35m 18s checked
  2014-02-06 10:16:25 "Summary: Total System Power 178W"
```

Sonexion® Administrator's Guide

```
snx11000n005-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"
snx11000n005-Enclosure-R1C1-5U "Power Statistics": OK for 28d 15h 44m 54s checked
  2014-02-06 10:12:42 "Summary: Total System Power 1061W"
[root@snx11000n000 ~]# cscli monitor elements -S sbb
snx11000n003-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"
snx11000n005-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@snx11000n000 ~]# cscli monitor elements -S volt
snx11000n003-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"
snx11000n005-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@snx11000n000 ~]# cscli monitor elements -S disk
snx11000n000 "Arrays and Disk Status": OK for 28d 15h 45m 0s checked 2014-02-06
  10:15:53 "All arrays are operating normally"
snx11000n001 "Arrays and Disk Status": OK for 28d 15h 46m 42s checked 2014-02-06
  10:13:07 "All arrays are operating normally"
snx11000n002 "Arrays and Disk Status": OK for 28d 15h 43m 24s checked 2014-02-06
  10:12:56 "All arrays are operating normally"
snx11000n003 "Arrays and Disk Status": OK for 28d 15h 43m 22s checked 2014-02-06
  10:16:24 "All arrays are operating normally"
snx11000n004 "Arrays and Disk Status": OK for 2d 14h 39m 21s checked 2014-02-06
  10:13:42 "All arrays are operating normally"
snx11000n005 "Arrays and Disk Status": OK for 17d 14h 30m 20s checked 2014-02-06
  10:12:41 "All arrays are operating normally"
[root@snx11000n000 ~]# cscli monitor elements -S arrays
snx11000n000 "Arrays and Disk Status": OK for 28d 15h 45m 10s checked 2014-02-06
  10:15:53 "All arrays are operating normally"
snx11000n001 "Arrays and Disk Status": OK for 28d 15h 46m 52s checked 2014-02-06
  10:13:07 "All arrays are operating normally"
snx11000n002 "Arrays and Disk Status": OK for 28d 15h 43m 34s checked 2014-02-06
  10:12:56 "All arrays are operating normally"
snx11000n003 "Arrays and Disk Status": OK for 28d 15h 43m 32s checked 2014-02-06
  10:16:24 "All arrays are operating normally"
snx11000n004 "Arrays and Disk Status": OK for 2d 14h 39m 31s checked 2014-02-06
  10:13:42 "All arrays are operating normally"
snx11000n005 "Arrays and Disk Status": OK for 17d 14h 30m 30s checked 2014-02-06
  10:12:41 "All arrays are operating normally"
[root@snx11000n000 ~]#
```

7.16.1 Manage the Netfilter Level

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]
```

Optional Arguments	Description
<code>-h --help</code>	Shows the help message and exits.
<code>-s --show</code>	Shows the current netfilter level.
<code>-l level --level level</code>	Sets the netfilter level (<code>off</code> , <code>lustre</code> , <code>on</code>).
<code>--force</code>	Forces the netfilter level to be set to <code>off</code> .

7.17 RAID Management

7.17.1 Enable RAID Checking

Modes: Daily

The `raid_check` command enables RAID check on RAID devices.

Synopsis

```
$ cscli raid_check -h (-a | -n node_list) [-i] [-c {on,off}] [--now]
[-s a_time]
```

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.
<code>-a --all</code>	Looks through all nodes elements.
<code>-n node_list</code> <code> --node node_list</code>	Looks through passed hostname elements. Looks for pdsh style nodes host names.
<code>-i --info</code>	Prints the current RAID check status for selected nodes.

-c {on,off} --cron {on,off}	Enables/Disables the cron job for the RAID check.
--now	Performs the RAID check now.
-s <i>a_time</i> --set <i>a_time</i>	Specifies a string to set a time to run the RAID check.

7.17.2 Manage RAID Rebuild Rate

Modes: Site configuration

The `rebuild_rate` command manages the RAID recovery rate management for data arrays. It sets the minimum rate in KB/s at which the system will perform recovery tasks after drive failure(s). The `rebuild_rate` command displays the numerical values when the rebuild is in progress. The minimum rates are now displayed on all nodes via `cscli rebuild_rate` command.

Synopsis

```
$ cscli rebuild_rate [-h] [-n nodes] [--reset] [-1 single_rate]
  [-m multiple_rate] [-b rebalance_rate]
```

Optional Arguments	Description
-h --help	Shows the help message and exits.
-n <i>nodes</i> --node <i>nodes</i>	Specifies pdsh-style node hostnames (e.g. node[100-110,120]). Sets global rates by omitting this parameter.
--reset	Reset rates to global values.
-1 <i>single_rate</i> --after-first-failure <i>single_rate</i>	After a single drive failure, use this rate to recover array redundancy. For RAID-6, this is the rebuild rate. For GridRAID, this is the reconstruction rate
-b <i>rebalance_rate</i> --rebalance <i>rebalance_rate</i>	For GridRAID, use this rate to rebalance data, parity, and spare space after reconstruction is completed for a single drive failure. For RAID-6, this parameter is ignored.
-m <i>multiple_rate</i> --after-multiple-failures <i>multiple_rate</i>	After multiple drive failure, increase to this rate. For RAID-6, this is the rebuild rate. For GridRAID, this is the reconstruction rate and the rebalance rate.

CLI Parameter	Diskmonitor
-1	Degraded parameter.
-b	Sets the diskmonitor with a new 'recover' parameter.
-m	Sets the "boost" rate for a -m parameter.

Example command with output:

```
[root@snx11000n000 ~]# cscli rebuild_rate -n snx11000 [200-202,204-207]
```

KB/Sec:

Node	Rebuild rate single	Rebalance rate	Rebuild rate multiple
snx11000n000	50000	50000	80000
snx11000n001	50000	50000	80000
snx11000n002	50000	50000	80000
snx11000n004	50000	50000	80000
snx11000n005	50000	50000	80000
snx11000n006	50000	50000	80000
snx11000n007	50000	50000	80000
global	50000	50000	80000

7.18 Administrative Settings

7.18.1 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.

7.18.2 Manage the System Date

The `set_date` command manages the date on the Sonexion system.

IMPORTANT: Exercise caution before using the `--force` parameter.

Synopsis

```
$ cscli set_date [-h] [-s new_date] [--force-ntp]
```

Optional Arguments	Description
<code>-h</code> <code>--help</code>	Displays the help message and exits.
<code>-s <i>new_date</i></code> <code>--set <i>new_date</i></code>	Specifies the new date in this format: <i>MMDDhhmmCCYY.ss</i> .
<code>--force-ntp</code>	Forces NTP configuration.

7.18.3 Manage the System Timezone

Modes: Daily

The `set_timezone` command manages the timezone on the Sonexion system.

Synopsis

```
$ cscli set_timezone [-h] [-s new_timezone] [-l]
```

Optional Arguments	Description
<code>-h</code> <code>--help</code>	Displays the help message and exits.
<code>-s <i>new_timezone</i></code> <code>--set <i>new_timezone</i></code>	Specifies the new time zone location name. For example, "America/Los_Angeles".
<code>-l</code> <code>--list</code>	Lists the available timezones.

NOTE: 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 on the Sonexion systems. However after using the `cscli set_timezone` command on the system, it does not get 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.

7.18.4 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 Sonexion 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
<code>-h</code> <code>--help</code>	Displays the help message and exits.
<code>-e</code> <code>--enable</code>	Enables the IB storage manager used with the Sonexion system.
<code>-d</code> <code>--disable</code>	Disables the IB storage manager used with the system.
<code>-P {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15}</code> <code>--priority</code> <code>{0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15}</code>	Sets the priority [0..15] of the IB storage manager used with <i>the</i> system.
<code>-c <i>cluster_name</i></code> <code>--cluster <i>cluster_name</i></code>	This parameter is deprecated. It is supported only for backward compatibility.

7.18.5 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.

NOTE: 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 allows to enable or disable a "guest" account access using UI or/and CSCLI and change "guests" 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 <i>password</i>	Guest account password.
--password-disable	Clears password and blocks account until a new password is set. NOTE: This option will also set “disabled” flags in cscli and GUI for guest account.

7.19 Managing Support Bundles

The `support_bundle` command manages support bundles and support bundle settings.

Synopsis

```
$ cscli support_bundle [-h] {collect,set,export,show}
```

Optional Arguments	Description
-h --help	Displays the help message and exits.
-c --collect-bundle	Collects the support bundle.
--show	Support bundle show command.
--export	Export support bundles as Tar.GZ archive (into the current folder).
--set	Support bundle set command.

Subcommand (support_bundle collect)

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
<code>-h --help</code>	Displays the help message and exits.
<code>-n <i>nodes</i></code> <code> --nodes <i>nodes</i></code>	Displays pdsh style node names. The default is all nodes.
<code>-t <i>minutes</i></code> <code> --time-window <i>minutes</i></code>	Displays the time window in minutes. The default is 45 minutes.

Subcommand (support_bundle show)

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
<code>-h --help</code>	Displays the help message and exits.
<code>-t --triggers</code>	Displays triggers that initiate automatic bundle collection.
<code>-b --bundles</code>	Displays a list of support bundles collected.
<code>-p --purge-limit</code>	Displays the purge limit. Free file system space limit in percents, after reaching which Trinity will purge old support bundle files.

Subcommand (support_bundle export)

The `support_bundle export` subcommand displays help, and the bundle ID.

Synopsis

```
$ cscli support_bundle export [-h] bundle_id
```

Positional Arguments	Description
<code>bundle_id</code>	Displays the support bundle ID.

Optional Arguments	Description
<code>-h --help</code>	Displays the help message and exits.

Subcommand (support_bundle set)

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
<code>-h --help</code>	Displays the help message and exits.
<code>-p <i>purge_limit</i></code> <code> --purge-limit <i>purge_limit</i></code>	Sets the purge-limit in percentage for support bundles.
<code>-t</code> <code> --trigger {<i>lbug</i>,Failover},</code> <code> {<i>lbug</i>,Failover}</code>	Displays trigger, that initiate automatic bundle collection.
<code>--on</code>	Turns the command on.
<code>--enable</code>	Enables the command.
<code>--off</code>	Turns the command off.
<code>--disable</code>	Disables the command.

8. GEM CLI Commands

This appendix details the supported command line interface (CLI) provided by the GEM software.

8.1 Serial port settings

Use the following settings for using HyperTerminal or other serial communications GUI to work with the CLI:

Baud rate (bits/sec):	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):	9600
Data bits:	8
Parity:	None
Stop bits:	1

Flow control: None

NOTE: 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

8.2 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

8.3 Supported commands

The following CLI commands are supported by the GEM software.

8.3.1 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, i.e. a simple single-shot diagnostic dump.
- Command arguments: None
- Command type: Diagnostic
- Access level: General

8.3.2 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

8.3.3 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

8.3.4 getvpd

Command name:	getvpd
Command synopsis:	Retrieves VPD information from all enclosure FRUs
Command description:	The getvpd command displays the following enclosure VPD data: <ul style="list-style-type: none"> • 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

8.3.5 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: 1 optional argument - see description above.
 Command type: Control
 Access level: General

8.3.6 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: ipmi_power [type]
 Type:

- 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

8.3.7 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

8.3.8 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

8.3.9 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

8.3.10 settime

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

8.3.11 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