Sun Network QDR InfiniBand Gateway Switch

Command Reference

Part No. 821-1189-11
June 2011, Revision A

Contents

Using This Documentation

This command reference provides information about the commands used to administer the Sun Network QDR InifiniBand Gateway Switch from Oracle. This document is written for technicians, system administrators, and users who have advanced experience administrating InfiniBand fabric hardware.

- "Related Documentation" on page xxi
- "Documentation, Support, and Training" on page xxiii

Related Documentation

The documents listed as online are available at:

(http://www.oracle.com/pls/topic/lookup?ctx=E19671-01&id=homepage)

Application	Title	Format	Location
Getting started	*Sun Network QDR Infiniband Gateway Switch Getting Started Guide*	Printed PDF	Shipping kit Online
Last-minute information	*Sun Network QDR Infiniband Gateway Switch Product Notes*	PDF	Online
Installation, administration, and service	*Sun Network QDR Infiniband Gateway Switch User's Guide*	PDF HTML	Online

Application	Title	Format	Location
Command reference	*Sun Network QDR Infiniband Gateway Switch Command Reference*	PDF HTML	Online
Compliance	*Sun Network QDR Infiniband Gateway Switch Safety and Compliance Guide*	PDF	Online
Oracle ILOM information	*Oracle Integrated Lights Out Manager (ILOM) 3.0 Supplement for the Sun Network QDR Infiniband Gateway Switch*	PDF HTML	Online

The Oracle ILOM 3.0 documents listed as online are available at:

(http://www.oracle.com/pls/topic/lookup?ctx=E19860-01&id=homepage)

Application	Title	Format	Location
Last-minute information	*Oracle Integrated Lights Out Manager (ILOM) 3.0 Feature Updates and Release Notes*	PDF HTML	Online
Getting started	*Oracle Integrated Lights Out Manager (ILOM) 3.0 Quick Start Guide*	PDF HTML	Online
Overview	*Oracle Integrated Lights Out Manager (ILOM) 3.0 Daily Management -- Concepts Guide*	PDF HTML	Online
Administration from web interface	*Oracle Integrated Lights Out Manager (ILOM) 3.0 Daily Management -- Web Procedures Guide*	PDF HTML	Online
Administration from CLI interface	*Oracle Integrated Lights Out Manager (ILOM) 3.0 Daily Management -- CLI Procedures Guide*	PDF HTML	Online
Administration from SNMP and other interfaces	*Oracle Integrated Lights Out Manager (ILOM) 3.0 Protocol Management -- SNMP, IPMI, CIM, WS-MAN*	PDF HTML	Online
Maintenance operations and diagnostics	*Oracle Integrated Lights Out Manager (ILOM) 3.0 Maintenance and Diagnostics -- CLI and Web Guide*	PDF HTML	Online
Remote redirection consoles	*Oracle Integrated Lights Out Manager (ILOM) 3.0 Remote Redirection Consoles -- CLI and Web Guide*	PDF HTML	Online

Documentation, Support, and Training

These web sites provide additional resources:

- Documentation (http://www.oracle.com/technetwork/indexes/documentation/index.html)
- Support (https://support.oracle.com)
- Training (https://education.oracle.com)

Understanding Hardware Commands

The hardware commands act upon or monitor the switch hardware. They are available through the Oracle ILOM CLI interface with the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` targets. See "Linux Shells for Hardware Commands" on page 3.

With the standard Linux shell CLI interface, only the `root` user of the management controller can run *all* of the hardware commands. The format of the hardware commands is as follows:

`#` *command* `[`*arguments*`][`*arguments*`]...`

Command Syntax	Links
`checkboot`	"checkboot Command" on page 6
`checkguidfilesftree [-h][-q][-v]`	"checkguidfilesftree Command" on page 7
`checkpower`	"checkpower Command" on page 8
`checktopomax [-h][-l][-v]`	"checktopomax Command" on page 9
`checkvoltages`	"checkvoltages Command" on page 11
`connector` *name* `present|portstate|info|dump [-h]`	"connector Command" on page 12
`createvlan` *connector* `-vlan` *vlan_ID* `-pkey` *p_key* `[-n][-h]`	"createvlan Command" on page 14
`createvnic` *connector* `<-guid` *guid*`>|<-host` *hostname* `-port` *port*`>|<-node` *string* `-port` *port*`> [-mac` *mac*`][-vlan` *vlan_ID*`] -pkey` *p_key* `[-n][-h]`	"createvnic Command" on page 15
`dcsport [-guid` *guid*`|-type DCS-gw -ibdev` *ibdev*`] -port` *port*`|-connector` *connector*`|-printconnectors|-printinternal`	"dcsport Command" on page 17
`deletevlan` *connector* `-vlan` *vlan_ID* `[-n][-h]`	"deletevlan Command" on page 19
`deletevnic` *connector VNIC_ID*	"deletevnic Command" on page 20
`disablecablelog`	"disablecablelog Command" on page 21
`disablegwport` *connector*	"disablegwport Command" on page 22

Command Syntax	Links
`disablelinklog`	"disablelinklog Command" on page 23
`disablesm`	"disablesm Command" on page 24
`disableswitchport [--reason=`*reason*`] `*connector*`|`*ibdev port*	"disableswitchport Command" on page 25
`disablevnic `*connector VNIC_ID* `[-h]`	"disablevnic Command" on page 27
`enablecablelog`	"enablecablelog Command" on page 28
`enablegwport `*connector*	"enablegwport Command" on page 28
`enablelinklog`	"enablelinklog Command" on page 29
`enablesm`	"enablesm Command" on page 30
`enableswitchport [--reason=`*reason*`] `*connector*`|`*ibdev port*	"enableswitchport Command" on page 31
`enablevnic `*connector VNIC_ID* `[-h]`	"enablevnic Command" on page 33
`env_test`	"env_test Command" on page 34
`exit`	"exit Command (Hardware)" on page 36
`generatetopology `*topofile*`|[-h]`	"generatetopology Command" on page 37
`getfanspeed`	"getfanspeed Command" on page 38
`getmaster [-l]`	"getmaster Command" on page 39
`getnm2type`	"getnm2type Command" on page 40
`getportcounters `*port*`|`*connector* `[-R]`	"getportcounters Command" on page 41
`getportstatus `*connector*`|`*ibdev port*	"getportstatus Command" on page 43
`help `*command*	*class*	"help Command (Hardware)" on page 45
`ibdevreset `*ibdev*	"ibdevreset Command" on page 46
`listlinkup`	"listlinkup Command" on page 47
`matchtopology `*topofile*`|[-h]`	"matchtopology Command" on page 49
`setcontrolledhandover `*state*`|list`	"setcontrolledhandover Command" on page 50
`setgwethport `*connector* `[-linkmode `*linkmode*`][-mtu `*mtu*`][-txpause `*pause*`][-rxpause `*pause*`][-h]`	"setgwethport Command" on page 52
`setgwinstance `*instance*`|list`	"setgwinstance Command" on page 54
`setgwsl eoib|ctrl `*level*	"setgwsl Command" on page 55
`setgwsystemname `*systemname*`|--list|--clear [-h]`	"setgwsystemname Command" on page 56
`setloghost `*IP_address*`|`*hostname*`|localhost`	"setloghost Command" on page 58
`setmsmlocationmonitor `*state*`|list [-h]`	"setmsmlocationmonitor Command" on page 59

Command Syntax	Links
`setsmpriority` *priority*|`list`	"setsmpriority Command" on page 60
`setsubnetprefix` *prefix*|`list`	"setsubnetprefix Command" on page 61
`showfruinfo`	"showfruinfo Command" on page 63
`showgwconfig`	"showgwconfig Command" on page 64
`showgwports [-v][-h]`	"showgwports Command" on page 65
`showioadapters`	"showioadapters Command" on page 67
`showpsufru` *slot*	"showpsufru Command" on page 69
`showsmlog [-h]`	"showsmlog Command" on page 70
`showtemps`	"showtemps Command" on page 71
`showtopology [-h]`	"showtopology Command" on page 72
`showunhealthy`	"showunhealthy Command" on page 75
`showvlan` *connector* `[-h]`	"showvlan Command" on page 76
`showvnics`	"showvnics Command" on page 78
`smconfigtest [-l][-h]`	"smconfigtest Command" on page 79
`version`	"version Command (Hardware)" on page 80

Related Information

- "Understanding InfiniBand Commands" on page 83
- "Understanding SNMP MIB OIDs" on page 153

Linux Shells for Hardware Commands

The preferred method of accessing the Linux shell is through the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Using the `show` command on the `/SYS/Switch_Diag` target opens a restricted Linux shell that enables the `ilom-admin` user, `ilom-operator` user, and users with similar permissions to run diagnostic commands.

Using the `show` command on the `/SYS/Gateway_Mgmt` target opens a different restricted Linux shell that enables the `ilom-admin` user and users with similar permissions to run both diagnostic and gateway management commands.

Using the `show` command on the `/SYS/Fabric_Mgmt` target opens still an other restricted Linux shell that enables the `ilom-admin` user and users with similar permissions to run diagnostic, gateway, and fabric management commands.

Note – The `ilom-operator` user cannot access the Linux shell from either the `/SYS/Gateway_Mgmt` or /SYS/Fabric_Mgmt target.

The following table lists the hardware commands and their availability from the respective Linux shell targets. Typing the `help all` command from within the restricted shells lists the commands available to that shell.

Note – Hardware commands that are not listed are unavailable from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, or `/SYS/Fabric_Mgmt` targets.

Command	/SYS/Switch_Diag	/SYS/Gateway_Mgmt	/SYS/Fabric_Mgmt
checkboot	Available	Available	Available
checkguidfilesftree			Available
checkpower	Available	Available	Available
checktopomax			Available
checkvoltages	Available	Available	Available
connector	Available	Available	Available
createvlan		Available	Available
createvnic		Available	Available
dcsport	Available	Available	Available
deletevlan		Available	Available
deletevnic		Available	Available
disablecablelog			Available
disablegwport		Available	Available
disablelinklog			Available
disablesm			Available
disableswitchport			Available
disablevnic		Available	Available
enablecablelog			Available
enablegwport		Available	Available
enablelinklog			Available

Command	/SYS/Switch_Diag	/SYS/Gateway_Mgmt	/SYS/Fabric_Mgmt
enablesm			Available
enableswitchport			Available
enablevnic		Available	Available
env_test	Available	Available	Available
exit	Available	Available	Available
generatetopology			Available
getfanspeed	Available	Available	Available
getmaster	Available	Available	Available
getportcounters	Available	Available	Available
getportstatus	Available	Available	Available
help	Available	Available	Available
listlinkup	Available	Available	Available
matchtopology			Available
setcontrolledhandover			Available
setgwethport		Available	Available
setgwinstance		Available	Available
setgwsl		Available	Available
setmsmlocationmonitor			Available
setsmpriority			Available
setsubnetprefix			Available
showfruinfo	Available	Available	Available
showgwconfig		Available	Available
showgwports		Available	Available
showpsufru	Available	Available	Available
showsmlog	Available	Available	Available
showtemps	Available	Available	Available
showtopology	Available	Available	Available
showunhealthy	Available	Available	Available
showvlan		Available	Available
showvnics		Available	Available
smconfigtest			Available

Command	/SYS/Switch_Diag	/SYS/Gateway_Mgmt	/SYS/Fabric_Mgmt
version	Available	Available	Available

Related Information

- "Linux Shells for InfiniBand Commands" on page 59

checkboot Command

Displays switch chip and BridgeX boot status.

Syntax

```
checkboot
```

Description

This hardware command checks the boot status of the switch chip and of the BridgeX chips. Output is a simplified OK for each.

The checkboot command is available from the /SYS/Switch_Diag, /SYS/Gateway_Mgmt, and /SYS/Fabric_Mgmt Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to check the boot status of the switch chip and BridgeX chips with the checkboot command.

```
# checkboot
Switch OK
Bridge-0 OK
Bridge-1 OK
#
```

Related Information

- "ibdevreset Command" on page 46
- "env_test Command" on page 34
- "Linux Shells for Hardware Commands" on page 3

`checkguidfilesftree` Command

Performs check of Ftree GUID files.

Syntax

```
checkguidfilesftree [-h][-q][-v]
```

Description

This hardware command is a script that performs a search for and check of the root node GUID file used for the Ftree routing algorithm. The command searches the `/etc/opensm/opensm.conf` file for the specified root node GUID filename and checks the integrity of that file. The file is typically the `/etc/opensm/guid.txt` file.

Note – This command is only useful for InfiniBand fabrics having non-homogenous Ftree topologies.

The `checkguidfilesftree` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `generic` command and their purposes:

Option	Purpose
`-h`	Provides help.
`-q`	Disables output unless errors are detected.
`-v`	Verbose mode.

Example

The following example shows how to check for the root node GUID file with the `checkguidfilesftree` command.

```
# checkguidfilesftree -q
#
```

Related Information

- "Linux Shells for Hardware Commands" on page 3

`checkpower` Command

Displays power supply status.

Syntax

```
checkpower
```

Description

This hardware command checks the status of the power supplies. Output is a simplified `OK`.

The `checkpower` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to check the power supply status with the `checkpower` command.

```
# checkpower
PSU 0 present status: OK
PSU 1 present status: OK
All PSUs OK
#
```

Related Information

- "checkvoltages Command" on page 11
- "env_test Command" on page 34
- "Linux Shells for Hardware Commands" on page 3

`checktopomax` Command

Displays size of topology.

Syntax

```
checktopomax [-h][-l][-v]
```

Description

This hardware command is a script that verifies if the number of switches and HCAs found in the InfiniBand fabric is less than the maximum values provided in the `/conf/dcsmonitor.conf` file.

The `checktopomax` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `checktopomax` command and their purposes:

Option	Purpose
`-h`	Provides help.
`-l`	Disables writing output to syslog.
`-v`	Verbose mode.

Example

The following example shows how to verify the number of switches and HCAs within your InfiniBand fabric with the `checktopomax` command.

Note – The output for your InfiniBand fabric will differ from that in the example.

```
# checktopomax -v
checktopomax: INFO: You have allowed max number of Switch elements to be 48
checktopomax: INFO: You have allowed max number of CA     elements to be 1500
checktopomax: INFO: Number of Switch elements empiric found by application
/usr/sbin/ibnodes: 2
checktopomax: INFO: Number of Ca     elements empiric found by application
/usr/sbin/ibnodes: 7
#
```

Related Information

- "ibnodes Command" on page 123
- "ibhosts Command" on page 116
- "Linux Shells for Hardware Commands" on page 3

`checkvoltages` Command

Verifies voltages.

Syntax

```
checkvoltages
```

Description

This hardware command displays the internal voltages for the main board. On the left side of the equals sign is the expected voltage. On the right side of the equals sign is the measured voltage. If the difference between the expected voltage and the measured voltage is more than 10%, the cause should be investigated. The command also provides a summary of the voltage conditions.

The `checkvoltages` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to check that voltages are at nominal values with the `checkvoltages` command.

```
# checkvoltages
Voltage ECB OK
Measured 3.3V Main = 3.30 V
Measured 3.3V Standby = 3.42 V
Measured 12V = 12.06 V
Measured 5V = 5.03 V
Measured VBAT = 3.17 V
Measured 1.0V = 1.01 V
Measured I4 1.2V = 1.22 V
Measured 2.5V = 2.51 V
Measured V1P2 DIG = 1.18 V
Measured V1P2 ANG = 1.18 V
Measured 1.2V BridgeX = 1.22 V
Measured 1.8V = 1.80 V
```

```
Measured 1.2V Standby = 1.20 V
All voltages OK
#
```

Related Information

- "checkpower Command" on page 8
- "env_test Command" on page 34
- "Linux Shells for Hardware Commands" on page 3

`connector` Command

Reads QSFP cable information.

Syntax

`connector` *name* `present|portstate|info|dump [-h]`

where *name* is the name of the connector (`0A – 15A`, `0A-ETH`, `1A-ETH`, `0B – 15B`).

Description

This hardware command performs a pass-fail test to verify that an InfiniBand cable is connected to a particular connector and to the switch chip port that the link routes. The command can also read the data registers of the cable and report FRU ID information.

The `connector` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `connector` command and their purposes:

Option	Purpose
`present`	Checks for the presence of connector *number*.
`portstate`	Returns a mapping of the connector to the respective port of the switch chip.
`info`	Displays FRU ID information.
`dump`	Displays a raw hexadecimal dump of the FRU ID information.
`-h`	Provides help.

Example

The following example shows how to display the FRU ID information for connector 1A with the `connector` command.

```
# connector 1A info
Cable connector 1A present
Indentifier: QSFP
Connector type: Copper pigtail
Vendor: Amphenol
Vendor OUI: 415048
Partnumber: 568400005
Revision: C
Serialnumber: APF08510050019
Date: 081219
#
```

Related Information

- "env_test Command" on page 34
- "listlinkup Command" on page 47
- "Linux Shells for Hardware Commands" on page 3

`createvlan` Command

Creates a VLAN.

Syntax

`createvlan` *connector* `-vlan` *vlan_ID* `-pkey` *p_key* `[-n][-h]`

where:

- *connector* is the name of the connector (`0A-ETH-1` to `0A-ETH-4`, `0A-ETH`, `1A-ETH-1` to `1A-ETH-4`, and `1A-ETH`).
- *vlan_ID* is the VLAN identifier.
- *p_key* is the partition key (`0` – `7fff` or `default`).

Description

This hardware command creates a VLAN association for a connector.

The `createvlan` command is available from the `/SYS/Gateway_Mgmt` and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `createvlan` command and their purposes:

Option	Purpose
`-vlan`	Specifies the VLAN identifier.
`-pkey`	Specifies the partition key value.
`-n`	Use for nonpersistent VLANs.
`-h`	Provides help.

Example

The following example shows how to create a VLAN association to connector 1A-ETH-3 with VLAN id `3` and pkey `5` using the `createvlan` command.

```
# createvlan 1A-ETH-3 -VLAN 3 -PKEY 5
#
```

Related Information

- "deletevlan Command" on page 19
- "showvlan Command" on page 76
- "Linux Shells for Hardware Commands" on page 3

`createvnic` Command

Creates a VNIC.

Syntax

`createvnic` *connector* `<-guid` *guid*`>|<-host` *hostname* `-port` *port*`>|<-node` *string* `-port` *port*`> [-mac` *mac*`][-vlan` *vlan_ID*`] -pkey` *p_key* `[-n][-h]`

where:

- *connector* is the name of the connector (`0A-ETH-1` to `0A-ETH-4`, `0A-ETH`, `1A-ETH-1` to `1A-ETH-4`, and `1A-ETH`).
- *guid* is the global unique identifier.
- *hostname* is the host name of the management controller or system name recognized by the BridgeX manager.
- *port* is the port of the host.
- *string* is the character string in the node description field.
- *mac* is the machine address code.

Note – Only even numbers are supported for the most significant byte of the MAC address (unicast).

- *vlan_ID* is the VLAN identifier.
- *p_key* is the partition key (`default`).

Description

This hardware command creates a VNIC for a specific connector. When a VNIC is created, it is automatically assigned an ID number. You can see this ID number with the `showvnics` command.

The `createvnic` command is available from the `/SYS/Gateway_Mgmt` and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `createvnic` command and their purposes:

Option	Purpose
`-guid`	Specifies the GUID.
`-host`	Specifies the host name or system name.
`-port`	Specifies the port of the host.
`-node`	Specifies the node description to identify the respective host.
`-mac`	Specifies the MAC address.
`-vlan`	Specifies the VLAN identifier.
`-pkey`	Specifies the partition key value.
`-n`	Use for nonpersistent VNICs.
`-h`	Provides help.

Example

The following example shows how to create a VNIC with the `default` partition key using the `createvnic` command.

```
# createvnic 1A-ETH-2 -GUID 00:21:28:00:01:3E:93:f7 -MAC 02:02:03:00:01:57 -pkey
default
VNIC created
#
```

Related Information

- "deletevnic Command" on page 20
- "disablevnic Command" on page 27
- "enablevnic Command" on page 33
- "showvnics Command" on page 78
- "Linux Shells for Hardware Commands" on page 3

`dcsport` Command

Maps between switch chip ports and QSFP connectors.

Syntax

`dcsport [-guid` *guid*`|-type DCS-gw -ibdev` *ibdev*`] -port` *port*`|-connector` *connector*`|-printconnectors|-printinternal`

where:

- *guid* is the global unique identifier.
- *ibdev* is the InfiniBand device name (`Switch`, `Bridge-0-0`, `Bridge-0-1`, `Bridge-1-0`, `Bridge-1-1`)
- *port* is the number of the port (`1–36`).
- *connector* is the name of the connector (`0A – 15A`, `0A-ETH`, `1A-ETH`, `0B – 15B`).

Note – The subordinate names of the 10GbE connectors, `0A-ETH-1` to `0A-ETH-4` and `1A-ETH-1` to `1A-ETH-4` are also valid *connector* names.

Description

This hardware command displays the mapping between I4 switch chip ports, BridgeX chip ports, and QSFP connectors. You can specify either a *port* or a *connector*.

The `dcsport` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `dcsport` command and their purposes:

Option	Purpose
`-guid`	Identifies the GUID of the IB device for mapping.
`-ibdev`	Identifies the name of the IB device for mappping.
`-port`	Identifies the *port* to provide the connector mapping.
`-connector`	Identifies the *connector* to provide the port mapping.
`-printconnectors`	Displays mapping for all connectors.
`-printinternal`	Displays I4 switch chip to BridgeX chip internal mapping.

Note – If no *guid* or *ibdev* are specified, the command defaults to the local I4 switch chip or BridgeX chips, inferred by the *port* number or *connector* name.

Example

The following example shows how to display the mapping for connector `0A-ETH` with the `dcsport` command.

```
# dcsport -connector 0A-ETH
Connector 0A-ETH maps to:
  0A-ETH-1 Bridge-0-1 port 0A-ETH-1
  0A-ETH-2 Bridge-0-1 port 0A-ETH-2
  0A-ETH-3 Bridge-0-0 port 0A-ETH-3
  0A-ETH-4 Bridge-0-0 port 0A-ETH-4
#
```

Related Information

- "enablesm Command" on page 30
- "setsmpriority Command" on page 60
- "Linux Shells for Hardware Commands" on page 3

`deletevlan` Command

Deletes a VLAN.

Syntax

`deletevlan` *connector* `-vlan` *vlan_ID* `[-n][-h]`

where:

- *connector* is the name of the connector (`0A-ETH-1` to `0A-ETH-4`, `0A-ETH`, `1A-ETH-1` to `1A-ETH-4`, and `1A-ETH`).
- *vlan_ID* is the VLAN identifier.

Description

This hardware command removes the association of a connector to a VLAN.

The `deletevlan` command is available from the `/SYS/Gateway_Mgmt` and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `deletevlan` command and their purposes:

Option	Purpose
`-vlan`	Specifies the VLAN identifier.
`-n`	Use for nonpersistent VLANs.
`-h`	Provides help.

Example

The following example shows how to delete a VLAN id `3` from connector `1A-ETH-3` with the `deletevlan` command.

```
# deletevlan 1A-ETH-3 -vlan 3
#
```

Related Information

- "createvlan Command" on page 14
- "showvlan Command" on page 76
- "Linux Shells for Hardware Commands" on page 3

`deletevnic` Command

Deletes a VNIC.

Syntax

`deletevnic` *connector VNIC_ID*

where:

- *connector* is the name of the connector (`0A-ETH-1` to `0A-ETH-4`, `0A-ETH`, `1A-ETH-1` to `1A-ETH-4`, and `1A-ETH`).
- *VNIC_ID* is the VNIC identifier.

Description

This hardware command deletes a VNIC from a connector.

The `deletevnic` command is available from the `/SYS/Gateway_Mgmt` and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to delete VNIC `1` on connector 0A-ETH-1 with the `deletevnic` command.

```
# deletevnic 0A-ETH-1 1
#
```

Related Information

- "createvnic Command" on page 15
- "disablevnic Command" on page 27
- "enablevnic Command" on page 33
- "showvnics Command" on page 78
- "Linux Shells for Hardware Commands" on page 3

`disablecablelog` Command

Disables logging of cable events.

Syntax

```
disablecablelog
```

Description

This hardware command disables logging of cable events to the Syslog.

The `disablecablelog` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Example

The following example shows how to disable logging of cable events to the Syslog with the `disablecablelog` command.

```
# disablecablelog
Stopping Environment daemon.                                [  OK  ]
Starting Environment daemon.                                [  OK  ]
#
```

Related Information

- "enablecablelog Command" on page 28
- "disablelinklog Command" on page 23
- "Linux Shells for Hardware Commands" on page 3

`disablegwport` Command

Disables a gateway port.

Syntax

`disablegwport` *connector*

where *connector* is the name of the connector hosting the port (`0A-ETH-1` to `0A-ETH-4`, `0A-ETH`, `1A-ETH-1` to `1A-ETH-4`, and `1A-ETH`).

Description

This hardware command disables a connector and its associated port on the gateway.

The `disablegwport` command is available from the `/SYS/Gateway_Mgmt` and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to disable the gateway port at connector `0A-ETH-3` with the `disablegwport` command.

```
# disablegwport 0A-ETH-3
#
```

Related Information

- "enablegwport Command" on page 28
- "disablevnic Command" on page 27
- "Linux Shells for Hardware Commands" on page 3

`disablelinklog` Command

Disables logging of link events.

Syntax

```
disablelinklog
```

Description

This hardware command disables logging of link events to the Syslog.

The `disablelinklog` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Example

The following example shows how to disable logging of link events to the Syslog with the `disablecablelog` command.

```
# disablelinklog
Stopping Environment daemon.                                [  OK  ]
Starting Environment daemon.                                [  OK  ]
#
```

Related Information

- "enablelinklog Command" on page 29
- "disablecablelog Command" on page 21
- "Linux Shells for Hardware Commands" on page 3

`disablesm` Command

Disables the Subnet Manager within the management controller.

Syntax

```
disablesm
```

Description

This hardware command disables the Subnet Manager within the management controller of the gateway. You use this command in the event that a Subnet Manager external to the gateway is preferred.

The `disablesm` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Example

The following example shows how to disable the Subnet Manager with the `disablesm` command.

```
# disablesm
Stopping partitiond daemon.                                [  OK  ]
Stopping IB Subnet Manager..                               [  OK  ]
#
```

Related Information

- "enablesm Command" on page 30
- "setsmpriority Command" on page 60
- "Linux Shells for Hardware Commands" on page 3

`disableswitchport` Command

Disables a connector or switch chip port.

Syntax

`disableswitchport [--reason=`*reason*`]` *connector*`|`*ibdev* *port*

where:

- *reason* is the reason for disabling the port, `Blacklist` or `Partition`.
- *connector* is the number of the QSFP connector (`0A–15B`).
- *ibdev* is the InfiniBand device name (`Switch`, `Bridge-0-0`, `Bridge-0-1`, `Bridge-1-0`, `Bridge-1-1`)
- *port* is the number of the port (`1–36`).

Description

This hardware command disables a QSFP connector and port on the switch chip or a port on the BridgeX chips. The command addresses either the connector or the port on the switch chip or the BridgeX port.

The `--reason` option enables you to use a passphrase to lock the state of the port:

- `Blacklist` – A connector and port pair are identified as being inaccessible because of unreliable operation.
- `Partition` – A connector and port pair are identified as being isolated from the InfiniBand fabric.

Both the `Blacklist` and `Partition` passphrases survive reboot. You unlock these passphrases using the `enableswitchport` command with the `--reason` option.

The `disableswitchport` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Example

The following example shows how to disable and blacklist connector 14A with the `disableswitchport` command.

```
# disableswitchport --reason=Blacklist 14A
Disable Switch port 7 reason: Blacklist
Initial PortInfo:
# Port info: DR path slid 65535; dlid 65535; 0 port 7
LinkState:.....Down
PhysLinkState:.....Polling
LinkWidthSupported:.....1X or 4X
LinkWidthEnabled:.....1X or 4X
LinkWidthActive:.....4X
LinkSpeedSupported:.....2.5 Gbps or 5.0 Gbps or 10.0 Gbps
LinkSpeedEnabled:.....2.5 Gbps or 5.0 Gbps or 10.0 Gbps
LinkSpeedActive:.....2.5 Gbps
After PortInfo set:
# Port info: DR path slid 65535; dlid 65535; 0 port 7
LinkState:.....Down
PhysLinkState:.....Disabled
#
```

Related Information

- "enableswitchport Command" on page 31
- "ibportstate Command" on page 125
- "getportstatus Command" on page 43
- "Linux Shells for Hardware Commands" on page 3

`disablevnic` Command

Disables a VNIC.

Syntax

`disablevnic` *connector VNIC_ID* `[-h]`

where:

- *connector* is the name of the connector (`0A-ETH-1` to `0A-ETH-4`, `0A-ETH`, `1A-ETH-1` to `1A-ETH-4`, and `1A-ETH`).
- *VNIC_ID* is the VNIC identifier.

Description

This hardware command disables a VNIC. The `-h` option provides help.

The `disablevnic` command is available from the `/SYS/Gateway_Mgmt` and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to disable VNIC `2` on connector `0A-ETH-2` with the `disablevnic` command.

```
# disablevnic 0A-ETH-2 2
#
```

Related Information

- "createvnic Command" on page 15
- "deletevnic Command" on page 20
- "enablevnic Command" on page 33
- "showvnics Command" on page 78
- "Linux Shells for Hardware Commands" on page 3

`enablecablelog` Command

Enables logging of cable events.

Syntax

```
enablecablelog
```

Description

This hardware command enables logging of cable events to the Syslog.

The `enablecablelog` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Example

The following example shows how to enable logging of cable events to the Syslog with the `enablecablelog` command.

```
# enablecablelog
Stopping Environment daemon.                                [  OK  ]
Starting Environment daemon.                                [  OK  ]
#
```

Related Information

- "enablecablelog Command" on page 28
- "enablelinklog Command" on page 29

`enablegwport` Command

Enables a gateway port.

Syntax

`enablegwport` *connector*

where *connector* is the name of the connector hosting the port (`0A-ETH-1` to `0A-ETH-4`, `0A-ETH`, `1A-ETH-1` to `1A-ETH-4`, and `1A-ETH`).

Description

This hardware command enables a previously disabled connector and its associated port on the gateway.

The `enablegwport` command is available from the `/SYS/Gateway_Mgmt` and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to enable the gateway port at connector `0A-ETH-3` with the `enablegwport` command.

```
# enablegwport 0A-ETH-3
#
```

Related Information

- "disablegwport Command" on page 22
- "enablevnic Command" on page 33
- "Linux Shells for Hardware Commands" on page 3

`enablelinklog` Command

Enables logging of link events.

Syntax

`enablelinklog`

Description

This hardware command enables logging of link events to the Syslog.

The `enablelinklog` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Example

The following example shows how to enable logging of link events to the Syslog with the `enablelinklog` command.

```
# enablelinklog
Stopping Environment daemon.                                [  OK  ]
Starting Environment daemon.                                [  OK  ]
#
```

Related Information

- "disablelinklog Command" on page 23
- "enablecablelog Command" on page 28
- "Linux Shells for Hardware Commands" on page 3

`enablesm` Command

Enables the Subnet Manager within the management controller.

Syntax

```
enablesm
```

Description

This hardware command enables the Subnet Manager on the management controller.

The `enablesm` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Example

The following example shows how to enable the Subnet Manager with the `enablesm` command.

```
# enablesm
Starting IB Subnet Manager.                                [  OK  ]
Starting partitiond daemon.                                [  OK  ]
#
```

Related Information

- "disablesm Command" on page 24
- "setsmpriority Command" on page 60
- "Linux Shells for Hardware Commands" on page 3

`enableswitchport` Command

Enables a connector or switch chip port.

Syntax

`enableswitchport [--reason=`*reason*`]` *connector*`|`*ibdev* *port*

where:

- *reason* is the reason for disabling the port, `Blacklist` or `Partition`.
- *connector* is the number of the QSFP connector (`0A`–`15B`).
- *ibdev* is the InfiniBand device name (`Switch`, `Bridge-0-0`, `Bridge-0-1`, `Bridge-1-0`, `Bridge-1-1`)
- *port* is the number of the port (`1`–`36`).

Description

This hardware command enables a QSFP connector and port on the switch chip or a port on the BridgeX chips. The command addresses either the connector or the port on the switch chip or the BridgeX port.

The `--reason` option enables you to use the `Blacklist` or `Partition` passphrases to unlock the state of the port as locked using the `disableswitchport` command.

The `enableswitchport` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Example

The following example shows how to enable and de-blacklist connector 14A with the `enableswitchport` command.

```
# enableswitchport --reason=Blacklist 14A
Enable Switch port 7
Initial PortInfo:
# Port info: DR path slid 65535; dlid 65535; 0 port 7
LinkState:.....Down
PhysLinkState:.....Disabled
LinkWidthSupported:.....1X or 4X
LinkWidthEnabled:.....1X or 4X
LinkWidthActive:.....4X
LinkSpeedSupported:.....2.5 Gbps or 5.0 Gbps or 10.0 Gbps
LinkSpeedEnabled:.....2.5 Gbps or 5.0 Gbps or 10.0 Gbps
LinkSpeedActive:.....2.5 Gbps
After PortInfo set:
# Port info: DR path slid 65535; dlid 65535; 0 port 7
LinkState:.....Down
PhysLinkState:.....Polling
#
```

Related Information

- "disableswitchport Command" on page 25
- "ibportstate Command" on page 125
- "getportstatus Command" on page 43
- "Linux Shells for Hardware Commands" on page 3

`enablevnic` Command

Enables a VNIC.

Syntax

`enablevnic` *connector VNIC_ID* `[-h]`

where:

- *connector* is the name of the connector (`0A-ETH-1` to `0A-ETH-4`, `0A-ETH`, `1A-ETH-1` to `1A-ETH-4`, and `1A-ETH`).
- *VNIC_ID* is the VNIC identifier.

Description

This hardware command enables a previously disabled VNIC. The `-h` option provides help.

The `enablevnic` command is available from the `/SYS/Gateway_Mgmt` and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to enable VNIC `2` on connector `0A-ETH-2` with the `enablevnic` command.

```
# enablevnic 0A-ETH-2 2
#
```

Related Information

- "createvnic Command" on page 15
- "deletevnic Command" on page 20
- "disablevnic Command" on page 27
- "showvnics Command" on page 78
- "Linux Shells for Hardware Commands" on page 3

env_test Command

Displays environmental status.

Syntax

```
env_test
```

Description

This hardware command performs a series of hardware and environmental tests of the gateway. This command is an amalgamation of the following commands:

- `checkpower`
- `checkvoltages`
- `showtemps`
- `getfanspeed`
- `connector`
- `checkboot`

The command output provides voltage and temperature values, pass-fail results, and error messages.

The `env_test` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to display the hardware and environmental status of the gateway with the `env_test` command.

```
# env_test
Environment test started:
Starting Voltage test:
Voltage ECB OK
Measured 3.3V Main = 3.28 V
Measured 3.3V Standby = 3.37 V
```

```
Measured 12V = 12.06 V
Measured 5V = 5.03 V
Measured VBAT = 3.25 V
Measured 1.0V = 1.01 V
Measured I4 1.2V = 1.22 V
Measured 2.5V = 2.51 V
Measured V1P2 DIG = 1.18 V
Measured V1P2 ANG = 1.18 V
Measured 1.2V BridgeX = 1.22 V
Measured 1.8V = 1.80 V
Measured 1.2V Standby = 1.20 V
Voltage test returned OK
Starting PSU test:
PSU 0 present
PSU 1 present
PSU test returned OK
Starting Temperature test:
Back temperature 30.5
Front temperature 30.6
SP temperature 34.1
Switch temperature 48, maxtemperature 49
Bridge-0 temperature 40, maxtemperature 41
Bridge-1 temperature 45, maxtemperature 45
Temperature test returned OK
Starting FAN test:
Fan 0 not present
Fan 1 running at rpm 12075
Fan 2 running at rpm 11960
Fan 3 running at rpm 12075
Fan 4 not present
FAN test returned OK
Starting Connector test:
Connector test returned OK
Starting Onboard ibdevice test:
Switch OK
Bridge-0 OK
Bridge-1 OK
All Internal ibdevices OK
Onboard ibdevice test returned OK
Environment test PASSED
#
```

Related Information

- "checkboot Command" on page 6
- "checkpower Command" on page 8
- "checkvoltages Command" on page 11

- "connector Command" on page 12
- "getfanspeed Command" on page 38
- "showtemps Command" on page 71
- "Linux Shells for Hardware Commands" on page 3

`exit` Command (Hardware)

Terminates the session.

Syntax

```
exit
```

Description

This hardware command immediately teminates the session with the management controller.

The `exit` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to terminate the session with the `exit` command.

```
# exit
Connection to 123.45.67.89 closed.
#
```

Related Information

- *Gateway Remote Management*, `exit` command

`generatetopology` Command

Creates topology file.

Syntax

`generatetopology` *topofile*|`[-h]`

where *topofile* is the filename and path to the topology file.

Description

This hardware command creates a topology file of the InfiniBand fabric. This file is not compatible with the topology file created by the `ibnetdiscover` command. The `generatetopology` command is used with the `matchtopology` and `showtopology` commands to determine changes in the InfiniBand fabric. The `-h` option provides help.

The `generatetopology` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Example

The following example shows how to create a topology file called `Topo.conf` with the `generatetopology` command.

```
# generatetopology AUG22.topo
It will take some time to generate a topology file. Please wait!
topo.conf file is created. Will now start generating the topo file
Wrote Topology file: AUG22.topo
#
```

The following is a portion of a topology file created by the `generatetopology` command.

Note – Your topology file will differ.

```
SUNIBQDRGW mnm34-98.example.com Bridge 0
  BX1/P1 -4x-10G-> I4_GENERIC mnm34-98 P4
     -4x-10G-> I4_GENERIC mnm34-98 P3
  BX3/P1 -4x-10G-> I4_GENERIC mnm34-98 P2
     -4x-10G-> I4_GENERIC mnm34-98 P1

DEV26428_02P mnm34-55_HCA-1
  P1 -4x-10G-> SUNIBQDRGW mnm34-97 C-7A
  P2 -4x-10G-> I4_GENERIC mnm34-98 P33

SUNIBQDRGW mnm34-97.example.com
  A-SW/P2 -4x-10G-> SUNIBQDRGW mnm34-97.example.com BX3/P1
  A-SW/P4 -4x-10G-> SUNIBQDRGW mnm34-97.example.com BX1/P1
  C-0B -4x-10G-> I4_GENERIC mnm34-98 P21
  C-0A -4x-10G-> I4_GENERIC mnm34-98 P20
  C-1B -4x-10G-> I4_GENERIC mnm34-98 P19
  C-1A -4x-10G-> I4_GENERIC mnm34-98 P22
  C-2B -4x-10G-> I4_GENERIC mnm34-98 P24
  C-2A -4x-10G-> I4_GENERIC mnm34-98 P23
.
.
.
DEV26428_02P mnm34-60_HCA-1
  P1 -4x-10G-> SUNIBQDRGW mnm34-97.example.com C-8A
  P2 -4x-10G-> I4_GENERIC mnm34-98 P31

# Created from mnm34-98 at Thu Aug 19 13:32:25 PDT 2010
```

Related Information

- "matchtopology Command" on page 49
- "showtopology Command" on page 72
- "ibnetdiscover Command" on page 118
- "Linux Shells for Hardware Commands" on page 3

`getfanspeed` Command

Displays fan speed.

Syntax

```
getfanspeed
```

Description

This hardware command displays the speed of the fans. The command also indicates if the fan is not present or has stopped.

The `getfanspeed` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to display fan speeds with the `getfanspeed` command.

```
# getfanspeed
Fan 0 not present
Fan 1 running at rpm 12075
Fan 2 running at rpm 11960
Fan 3 running at rpm 12075
Fan 4 not present
#
```

Related Information

- "env_test Command" on page 34
- "Linux Shells for Hardware Commands" on page 3

`getmaster` Command

Displays primary (or Master) Subnet Manager node information.

Syntax

`getmaster [-l]`

Description

This hardware command returns information about the node that hosts the primary (or master) Subnet Manager of the InfiniBand fabric. The `-l` option provides a short historical list of Subnet Manager activity.

The `getmaster` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to display information about the node that hosts the master Subnet Manager with the `getmaster` command.

```
# getmaster -l
Local SM enabled and running
Last ring buffer history listed:
20100927 19:16:12 whereismaster started
20100927 19:16:12 Master SubnetManager on sm lid 0 sm guid 0x212856cfe2c0a0 :
20100927 19:16:29 Master SubnetManager on sm lid 6 sm guid 0x212856cfe2c0a0 :
SUN IB QDR GW switch mnm34-98.example.com
#
```

Related Information

- "sminfo Command" on page 146
- "Linux Shells for Hardware Commands" on page 3

`getnm2type` Command

Displays the device type.

Syntax

```
getnm2type
```

Description

This hardware command displays the type of InfiniBand device the management controller is installed within. The output is:

- `gw` – Sun Network QDR InfiniBand Gateway Switch
- `36p` – Sun Datacenter InfiniBand Switch 36
- `72p` – Sun Datacenter InfiniBand Switch 72

Example

The following example shows how to display the type of InfiniBand device with the `getnm2type` command.

```
# getnm2type
gw
#
```

Related Information

- "version Command (Hardware)" on page 80

`getportcounters` Command

Displays port counters.

Syntax

`getportcounters` *port*|*connector* `[-R]`

where:

- *port* is the number of the port (`1–36`).

- *connector* is the name of the connector (`0A` – `15A`, `0A-ETH`, `1A-ETH`, `0B` – `15B`).

Note – The subordinate names of the 10 GbE connectors, `0A-ETH-1` to `0A-ETH-4` and `1A-ETH-1` to `1A-ETH-4` are also valid *connector* names.

Description

This hardware command returns the error and data throughput counters of the specified *port* of the switch chip or *connector* of the gateway. If a *port* is specified, then the output is for InfiniBand counters. Similarly, if the *connector* specified is `0A` to `15B`, the output is also for InfiniBand counters. If the *connector* specified is `0A-ETH`, `1A-ETH`, or a subordinate connector, the output is for 10GbE counters.

The `-R` option clears the counters for the specified *port* or *connector*.

The `getportcounters` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to display the status of connector `1A-ETH-1` with the `getportcounters` command.

```
# getportcounters 1A-ETH-1
ETH Port 1A-ETH-1
---------------------------------
RX bytes:.......................0x6a5559
RX packets:.....................0xd650
RX Jumbo packets:...............0x0
RX unicast packets:.............0x0
RX multicast packets:...........0xbcfc
RX broadcast packets:...........0x1954
RX no buffer:...................0x0
RX CRC:.........................0x0
RX runt:........................0x0
RX errors:......................0x0
TX bytes:.......................0x0
TX packets:.....................0x0
TX Jumbo packets:...............0x0
TX unicast packets:.............0x0
TX multicast packets:...........0x0
```

```
TX broadcast packets:.....0x0
TX errors:.....0x0
#
```

Related Information

- "ibcheckport Command" on page 93
- "ibcheckportstate Command" on page 95
- "ibcheckportwidth Command" on page 97

getportstatus Command

Displays port status.

Syntax

`getportstatus` *connector*|*ibdev* *port*

where:

- *connector* is the name of the connector (`0A – 15A`, `0A-ETH`, `1A-ETH`, `0B – 15B`).

Note – The subordinate names of the 10GbE connectors, `0A-ETH-1` to `0A-ETH-4` and `1A-ETH-1` to `1A-ETH-4`, are also valid *connector* names.

- *ibdev* is the InfiniBand device name (`Switch`, `Bridge-0-0`, `Bridge-0-1`, `Bridge-1-0`, `Bridge-1-1`)
- *port* is the number of the port (`1–36`).

Description

This hardware command returns the status of the specified *connector* or *port* of the switch chip.

The `getportstatus` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to display the status of connector 9A with the `getportstatus` command.

Note – The parameters with the string `Active` indicate the current conditions.

```
# getportstatus 9A
Port status for connector 9A Switch Port 14
Adminstate:.....................Enabled
LinkWidthEnabled:...............1X or 4X
LinkWidthSupported:.............1X or 4X
LinkWidthActive:................4X
LinkSpeedSupported:.............2.5 Gbps or 5.0 Gbps or 10.0 Gbps
LinkState:......................Active
PhysLinkState:..................LinkUp
LinkSpeedActive:................10.0 Gbps
LinkSpeedEnabled:...............2.5 Gbps or 5.0 Gbps or 10.0 Gbps
#
```

The following example shows how to display the status of connector 0A-ETH-1 with the `getportstatus` command.

```
# getportstatus 0A-ETH-1
Port status for connector 0A-ETH-1 Bridge-0 Port Bridge-0-2
Adminstate.....................Enabled
State..........................Up
Link state.....................Up
Protocol.......................Ethernet
Link mode......................XFI
Speed..........................10Gb/s
MTU............................9600
Tx pause.......................Global
Rx pause.......................Global
#
```

Related Information

- "ibcheckport Command" on page 93
- "ibcheckportstate Command" on page 95
- "ibcheckportwidth Command" on page 97

`help` Command (Hardware)

Displays help for a command.

Syntax

`help` *command* | *class*

where:

- *command* is the command for which you need help.
- *class* is the catagory of commands.

Description

This hardware command provides help information for supported commands. Typically, the same information is provided using the `-h` option.

The `help` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to get help with the `help` command.

```
# help
List of avaliable classes:
general  -- General commands
diag     -- Switch diagnostics commands
ibdiag   -- Infiniband diagnostics commands
sm       -- Subnet Managager administration commands
other    -- Other managagement commands
all      -- Show all commands
Type "help" followed by a class name for a list of commands in that class.
Type "help" followed by a command name for description and usage of the command.
#
```

Related Information

- *Gateway Remote Management*, `help` command
- "Linux Shells for Hardware Commands" on page 3

`ibdevreset` Command

Resets the switch chip.

Syntax

`ibdevreset` *ibdev*

where *ibdev* is the InfiniBand device name (`Switch`, `Bridge-0`, or `Bridge-1`).

Description

This hardware command resets the InfiniBand device.

Example

The following example shows how to reset the switch chip with the `ibdevreset` command.

```
# ibdevreset Switch
Stopping IB Subnet Manager..                               [  OK  ]
Stopping whereismaster daemon.                             [  OK  ]
Stopping Environment daemon.                               [  OK  ]
Resetting Switch
Starting Environment daemon.                               [  OK  ]
Starting whereismaster daemon.                             [  OK  ]
Starting IB Subnet Manager.                                [  OK  ]
#
```

Related Information

- "checkboot Command" on page 6

`listlinkup` Command

Displays links presence.

Syntax

```
listlinkup
```

Description

This hardware command lists the presence of links and the up-down state of the associated ports on the switch chip.

The `listlinkup` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to display link presence and associated ports with the `listlinkup` command.

```
# listlinkup
Connector  0A Present <-> Switch Port 20 up (Enabled)
Connector  1A Present <-> Switch Port 22 up (Enabled)
Connector  2A Present <-> Switch Port 24 up (Enabled)
.
.
.
Connector 15A Not present
Connector 0A-ETH Present
 Bridge-0-1 Port 0A-ETH-1 up (Enabled)
 Bridge-0-1 Port 0A-ETH-2 up (Enabled)
 Bridge-0-0 Port 0A-ETH-3 up (Enabled)
 Bridge-0-0 Port 0A-ETH-4 up (Enabled)
Connector 1A-ETH Present
```

```
 Bridge-1-1 Port 1A-ETH-1 up (Enabled)
 Bridge-1-1 Port 1A-ETH-2 up (Enabled)
 Bridge-1-0 Port 1A-ETH-3 up (Enabled)
 Bridge-1-0 Port 1A-ETH-4 up (Enabled)
Connector  0B Present <-> Switch Port 19 up (Enabled)
Connector  1B Present <-> Switch Port 21 up (Enabled)
.
.
.
Connector 15B Not present
#
```

Related Information

- "ibportstate Command" on page 125
- "Linux Shells for Hardware Commands" on page 3

managementreset Command

Resets the management controller.

Syntax

```
managementreset [-r]
```

Description

This hardware command resets the management controller, the CPLD, and the I4 switch chip. The command requests a reboot, and in most situations, this should be done. The `managementreset` command also forces links to retrain upon management controller services startup. The `-r` option bypasses the query for reboot and reboots automatically.

Example

The following example shows how to reset the management controller and InfiniBand fabric services with the `managementreset` command.

Note – By rebooting the management controller, the link to the management console is severed. You must re-access the management controller to regain administrative control.

```
# managementreset
Stopping Environment daemon, please wait
Resetting CPLD, please wait
Restarting Environment daemon
Reboot needed to reconnect to I4 and enable IB ports
Do you want do reboot now [yes/no]:yes
Broadcast message from root (pts/0) (Fri Nov 20 17:10:27 2009):
The system is going down for reboot NOW!
# Connection to 123.45.67.89 closed by remote host.
Connection to 123.45.67.89 closed.
#
```

Related Information

- "ibdevreset Command" on page 46
- "ibportstate Command" on page 125

matchtopology Command

Compares topology file to InfiniBand fabric.

Syntax

`matchtopology` *topofile*|`[-h]`

where *topofile* is the filename and path to the topology file.

Description

This hardware command compares the topology file created with the `generatetopology` command with the current InfiniBand fabric topology. An error is displayed upon mismatch. The `-h` option provides help.

The `matchtopology` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Example

The following example shows how to compare the topology file to the current topology with the `matchtopology` command.

```
# matchtopology AUG22.topo
Topology matching will take some time. Please wait!
-I-------------------------------------------------
-I- Topology matching results
-I-------------------------------------------------
-I- The topology defined in topo1.topo perfectly matches the
discoverd fabric.
---------------------------------------------------------------------
#
```

Related Information

- "generatetopology Command" on page 37
- "showtopology Command" on page 72
- "Linux Shells for Hardware Commands" on page 3

`setcontrolledhandover` Command

Enables or disables controlled handover.

Syntax

`setcontrolledhandover` *state*|`list`

where *state* is either `TRUE`, or `FALSE`.

Description

This hardware command enables or disables the controlled handover feature. The `TRUE` option enables and the `FALSE` option disables the action. The `list` option provides a listing of the Subnet Manager settings in respect to priority, controlled handover, and subnet prefix.

Note – You must stop or disable the OpenSM Subnet Manager before using the `setcontrolledhandover` command. See "disablesm Command" on page 24.

The InfiniBand specification identifies two parameters used to negotiate a master Subnet Manager - the Subnet Manager priority and the port GUID. Should two or more Subnet Managers have the same highest configured priority, the Subnet Manager with the lowest GUID becomes the master Subnet Manager.

If an un-controlled or automatic handover state exists, when the primary-and-master Subnet Manager fails, the secondary-and-standby Subnet Manager takes on the role of the new master Subnet Manager. During this time, traffic is temporarily suspended as the secondary-and-master Subnet Manager discovers and sets up the InfiniBand fabric.

If the primary Subnet Manager is able to recover from the failure, reboot, and reinitialize itself, it identifies itself as the intended master Subnet Manager. Again, traffic is temporarily suspended as the primary-and-master Subnet Manager re-discovers and sets up the InfiniBand fabric. The secondary Subnet Manager returns to the standby status.

If controlled handover is enabled, then some priority values become reserved. The Subnet Managers dynamically adjust their respective priorities to avoid the dual handover situation. A secondary Subnet Manager that became a master Subnet Manager due to a handover raises its priority to a reserved value and retains its master Subnet Manager status. Regardless if the primary Subnet Manager comes back online, the second handover does not occur.

The user-configured priorities of the Subnet Managers are retained, only during a handover are reserved priorities used.

The result of this scheme is that the user-configured priority is respected during system boot, but dual handover does not occur because the primary Subnet Manager loses its master Subnet Manager status (secondary Subnet Manager priority changes to reserve).

Note – Manually starting Subnet Managers individually is equivalent to the system boot sequence.

The `setcontrolledhandover` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Example

The following example shows how to enable a controlled handover of Subnet Managers with the `setcontrolledhandover` command.

```
# setcontrolledhandover TRUE
Current SM setttings:
smpriority 0
controlled_handover TRUE
subnet_prefix 0x fe80000000000000
#
```

Related Information

- "enablesm Command" on page 30
- "setsmpriority Command" on page 60
- "setsubnetprefix Command" on page 61
- "Linux Shells for Hardware Commands" on page 3

`setgwethport` Command

Configures an Ethernet port.

Syntax

`setgwethport` *connector* `[-linkmode` *linkmode*`][-mtu` *mtu*`][-txpause` *pause*`][-rxpause` *pause*`][-h]`

where:

- *connector* is the name of the connector (`0A-ETH` or `1A-ETH`).
- *linkmode* is the is the link mode, `XFI` (10Gb/s).
- *mtu* is the value of the MTU.
- *pause* is the priority flow control pause in quanta (`0– 255`, `Global`, or `None`)

Description

This hardware command manually configures an Ethernet port of the gateway. The Ethernet ports are set to 10Gb/s by default. Should a slower Ethernet device be attached to the connector, there is no auto-negotiation. Using the `setgwethport` command without options returns the current Ethernet status of the connector.

The `setgwethport` command is available from the `/SYS/Gateway_Mgmt` and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `setgwethport` command and their purposes:

Option	Purpose
`-linkmode`	Specifies the protocol, and consequent the link speed.
`-mtu`	Specifies the MTU.
`-txpause`	Specifies the transmit pause.
`-rxpause`	Specifies the receive pause.
`-h`	Provides help.

Example

The following example shows how to set the speed of connector `1A-ETH` to 10 GB/s using the `setgwethport` command.

```
# setgwethport 1A-ETH -linkmode XFI
Port status for connector 0A-ETH-1:
Adminstate.....................Enabled
State..........................Up
Link state.....................Up
Protocol.......................Ethernet
Link Mode......................XFI
Speed..........................10Gb/s
MTU............................9600
Tx pause.......................Global
Rx pause.......................Global
Port status for connector 1A-ETH-2:
.
```

```
.
.
Port status for connector 1A-ETH-4:
Adminstate.....................Enabled
State..........................Up
Link state.....................Up
Protocol.......................Ethernet
Link Mode......................XFI
Speed..........................10Gb/s
MTU............................9600
Tx pause.......................Global
Rx pause.......................Global
#
```

Related Information

- "getportstatus Command" on page 43
- "Linux Shells for Hardware Commands" on page 3

`setgwinstance` Command

Sets the gateway instance.

Syntax

`setgwinstance` *instance*|`list`

where *instance* is a number from `0` to `63`.

Description

When you configure multiple gateways, the gateways identify themselves to each other with a gateway instance number. No two gateways can have the same instance number.

Note – The `setgwinstance` command automatically stops and restarts the BridgeX manager.

Using the `list` option displays known gateways and their instance numbers.

The `setgwinstance` command is available from the `/SYS/Gateway_Mgmt` and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to set the gateway instance of the gateway to 3 using the `setgwinstance` command.

```
# setgwinstance 3
Stopping Bridge Manager..                                  [  OK  ]
Starting Bridge Manager.                                   [  OK  ]
#
```

The following example shows how to display the gateway instances of known gateways with the `list` option of the `setgwinstance` command.

```
# setgwinstance list
Current GW instance: 3
#
```

Related Information

- "Linux Shells for Hardware Commands" on page 3

`setgwsl` Command

Sets the service level.

Syntax

`setgwsl eoib|ctrl` *level*

where *level* is the service level from 0 to 15.

Description

This hardware command sets the service level of the Ethernet over InfiniBand (EoIB) data traffic (`eoib`) or control traffic (`ctrl`). The `showgwconfig` command displays the current service levels.

The `setgwsl` command is available from the `/SYS/Gateway_Mgmt` and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to set the EoIB data traffic service level to 2 using the `setgwsl` command.

```
# setgwsl eoib 2
Stopping Bridge Manager..                                  [  OK  ]
Starting Bridge Manager.                                   [  OK  ]
#
```

Related Information

- "showgwconfig Command" on page 64
- "Linux Shells for Hardware Commands" on page 3

`setgwsystemname` Command

Sets the system name.

Syntax

`setgwsystemname` *systemname*`|--list|--clear [-h]`

where *systemname* is the BridgeX manager's system name.

Description

This hardware command displays or sets the system name as recognized by the BridgeX manager. By default, the host name is the system name. The system name cannot be more than 19 characters long, otherwise it is truncated to 19 characters.

Options

The following table describes the options to the `setgwsystemname` command and their purposes:

Option	Purpose
`--clear`	Clears the configured string and reverts back to the host name.
`--list`	Displays the current string.
`-h`	Provides help.

Note – If you want to use a double-dash (--) in the system name, you must delimit the string with escape characters. See the example.

Example

The following example shows how to set the system name recognized by the BridgeX manager to `--myname` using the `setgwsystemname` command.

```
# setgwsystemname "\"--myname\""
Stopping Bridge Manager..                                  [  OK  ]
Starting Bridge Manager.                                   [  OK  ]
# setgwsystemname --list
BXM system name set to --myname
#
```

Related Information

-
- "Linux Shells for Hardware Commands" on page 3

`setloghost` Command

Sets the remote log host.

Syntax

`setloghost` *IP_address*|*hostname*|`localhost`

where:

- *IP_address* is the IP address of the remote server hosting the Syslog.
- *hostname* is the host name of the remote server hosting the Syslog.

Description

This hardware command sets the remote location to where the Syslog output is directed. The `localhost` option disables the redirection.

Example

The following example shows how to redirect the ouput of Syslog to the server `SyslogHost` using the `setloghost` command.

Note – You *must* restart the Subnet Manager using the `disablesm` and `enablesm` commands after setting the loghost.

```
# setloghost SyslogHost
Shutting down kernel logger:                               [  OK  ]
Shutting down system logger:                               [  OK  ]
Starting system logger:                                    [  OK  ]
Starting kernel logger:                                    [  OK  ]
#
```

Related Information

- "disablesm Command" on page 24
- "enablesm Command" on page 30

`setmsmlocationmonitor` Command

Sets monitoring of the Subnet Manager.

Syntax

`setmsmlocationmonitor` *state*|`list [-h]`

where *state* is either `enable` or `disable`.

Description

This hardware command sets the state of monitoring the location and condition of the primary (or Master) Subnet Manager for the InfiniBand fabric. You can set the state of monitoring to either `enabled` or `disabled`. The `list` option displays the current state of monitoring.

Example

The following example shows how to display the state of the Subnet Manager monitor with the `setmsmlocationmonitor` command.

```
# setmsmlocationmonitor list
Current state of Master Subnet Manager monitioring: enable
#
```

Related Information

- "disablesm Command" on page 24
- "enablesm Command" on page 30

`setsmpriority` Command

Sets the Subnet Manager priority.

Syntax

`setsmpriority` *priority*|`list`

where *priority* is a number from `0` (lowest) to `13` (highest).

Description

This hardware command sets the priority of the Subnet Manager within the management controller. You use this command when there are multiple Subnet Managers in the InfiniBand fabric. By setting a Subnet Manager to a higher priority than another Subnet Manager, it becomes the primary (or Master) Subnet Manager. When you set a Subnet Manager to a priority lower than another Subnet Manager, it becomes the secondary Subnet Manager. The `list` option provides a listing of the Subnet Manager settings in respect to priority, controlled handover, and subnet prefix.

The `setsmpriority` command writes the value of *priority* to the `sm_priority` parameter of the `/etc/opensm/opensm.conf` file.

Note – You must stop or disable the OpenSM Subnet Manager before using the `setsmpriority` command. See "disablesm Command" on page 24.

Note – Setting Subnet Managers of the same fabric to the same priority can have undesirable results.

The `setsmpriority` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Example

The following example shows how to set the priority of the Subnet Manager to 3 using the `setsmpriority` command.

```
# setsmpriority 3
-----------------------------------------------
OpenSM 3.2.6_20090717
  Reading Cached Option File: /etc/opensm/opensm.conf
  Loading Cached Option:routing_engine = ftree
  Loading Cached Option:sm_priority = 13
  Loading Cached Option:sminfo_polling_timeout = 1000
  Loading Cached Option:polling_retry_number = 3
Command Line Arguments:
  Priority = 3
  Creating config file template '/tmp/osm.conf'.
  Log File: /var/log/opensm.log
-----------------------------------------------
#
```

The following example shows how to display the current settings of the Subnet Manager priority, controlled handover, and subnet prefix with the `list` option of the `setsubnetprefix` command.

```
# setsmpriority list
smpriority 0
controlled_handover FALSE
subnet_prefix 0xfe80000000000000
#
```

Related Information

- "enablesm Command" on page 30
- "setcontrolledhandover Command" on page 50
- "setsubnetprefix Command" on page 61
- "Linux Shells for Hardware Commands" on page 3

`setsubnetprefix` Command

Sets the subnet prefix.

Syntax

`setsubnetprefix` *prefix*|`list`

where *prefix* is the hexadecimal prefix.

Description

This hardware command sets the subnet prefix for the InfiniBand fabric. The prefix is entered in hexadecimal, starting with `0x` and having lower case alphanumeric characters. The `list` option provides a listing of the Subnet Manager settings in respect to priority, controlled handover, and subnet prefix.

The `setsubnetprefix` command writes the value of *prefix* to the `subnet_prefix` parameter of the `/etc/opensm/opensm.conf` file.

Note – You must stop or disable the OpenSM Subnet Manager before using the `setsubnetprefix` command. See "disablesm Command" on page 24.

The `setsubnetprefix` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Example

The following example shows how to set the subnet prefix with the `setsubnetprefix` command.

```
# setsubnetprefix 0xabbababe
#
```

Related Information

- "enablesm Command" on page 30
- "setcontrolledhandover Command" on page 50
- "setsmpriority Command" on page 60
- "Linux Shells for Hardware Commands" on page 3

`showfruinfo` Command

Displays chassis FRU information.

Syntax

```
showfruinfo
```

Description

This hardware command displays the contents of the chassis FRU ID registers.

The `showfruinfo` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to display the chassis FRU ID registers with the `showfruinfo` command.

```
# showfruinfo
 UNIX_Timestamp32              :  Tue Mar 23 17:02:15 2010
 Sun_Fru_Description           :  ASSY,NM2-GW
 Vendor_ID_Code                :  11 E1
 Vendor_ID_Code_Source         :  01
 Vendor_Name_And_Site_Location :  4577 CELESTICA CORP. SAN JOSE CA US
 Sun_Part_Number               :  5111402
 Sun_Serial_Number             :  0110SJC-1010NG0045
 Serial_Number_Format          :  4V3F1-2Y2W2X4S
 Initial_HW_Dash_Level         :  03
 Initial_HW_Rev_Level          :  50
 Sun_Fru_Shortname             :  NM2 gateway
 Sun_Hazard_Class_Code         :  Y
 Sun_SpecPartNo                :  885-1655-01
#
```

Related Information

- "showpsufru Command" on page 69
- "Linux Shells for Hardware Commands" on page 3

showgwconfig Command

Displays gateway configuration.

Syntax

```
showgwconfig
```

Description

This hardware command displays status, version, and operational information for the gateway. The output includes a table of configuration parameters, and indicates if those parameters have been configured or are using default values.

The `showgwconfig` command is available from the `/SYS/Gateway_Mgmt` and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to display status, version, and operational information for the gateway using the `showconfig` command.

```
# showgwconfig
BXM (pid 17030) is running
BXM versions: bxm_user 1.3.6-0, BXM-API 1.6.0, bxm_libs 1.3.6-0, bxm_main 1.30
mlx_bx_core 1.30
Parameter        Configured Value     Running Value
-------------------------------------------------
GWInstance       None                 34
SystemName       None                 gw-2
EoIB Data SL     1                    1
EoIB Control SL  1                    1
#
```

Related Information

- "disablesm Command" on page 24
- "enablesm Command" on page 30
- "Linux Shells for Hardware Commands" on page 3

`showgwports` Command

Displays gateway port status.

Syntax

```
showgwports [-v][-h]
```

Description

This hardware command displays information about the gateway Ethernet ports. The information includes:

- GUIDs
- Node descriptions and names
- Ports
- LIDs
- States
- Modes
- Speed
- MTU

The `showgwports` command is available from the `/SYS/Gateway_Mgmt` and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `showgwports` command and their purposes:

Option	Purpose
`-v`	Displays verbose output.
`-h`	Provides help.

Example

The following example shows how to display verbose information about the gateway Ethernet ports using the `showgwports` command.

Note – Your output will differ from that of the example.

```
# showgwports -v

BRIDGE DEVICES:
---------------
NodeGUID           NodeDescrition
---------------------------------
0x0021283bad45c000 SUN IB QDR GW switch gw-2.example.com Bridge 0
0x0021283bad45c040 SUN IB QDR GW switch gw-2.example.com Bridge 1

INTERNAL PORTS:
---------------
Device   Port Portname  PeerPort PortGUID          LID    IBState  GWState
Speed  VLs MTU
-------------------------------------------------------------------------------
-----------------
Bridge-0 1  Bridge-0-1   4   0x0021283bad45c001 0x0007 Active  Up       40Gbs
2 4096
Bridge-0 2  Bridge-0-2   3   0x0021283bad45c002 0x0008 Active  Up       40Gbs
2 4096
Bridge-1 1  Bridge-1-1   2   0x0021283bad45c041 0x0009 Active  Up       40Gbs
2 4096
Bridge-1 2  Bridge-1-2   1   0x0021283bad45c042 0x000a Active  Up       40Gbs
2 4096

CONNECTOR 0A-ETH:
-----------------
```

```
Port      Bridge      Adminstate Link  State       Linkmode       Speed   MTU
TxPause  RxPause
-------------------------------------------------------------------------------
-------------------
0A-ETH-1  Bridge-0-2  Enabled    Up    Up          XFI            10Gb/s  9600
Global   Global
0A-ETH-2  Bridge-0-2  Enabled    Up    Up          XFI            10Gb/s  9600
Global   Global
0A-ETH-3  Bridge-0-1  Enabled    Up    Up          XFI            10Gb/s  9600
Global   Global
0A-ETH-4  Bridge-0-1  Enabled    Up    Up          XFI            10Gb/s  9600
Global   Global

CONNECTOR 1A-ETH:
-----------------
Port      Bridge      Adminstate Link  State       Linkmode       Speed   MTU
TxPause  RxPause
-------------------------------------------------------------------------------
-------------------
1A-ETH-1  Bridge-1-2  Enabled    Up    Up          XFI            10Gb/s  9600
Global   Global
1A-ETH-2  Bridge-1-2  Enabled    Up    Up          XFI            10Gb/s  9600
Global   Global
1A-ETH-3  Bridge-1-1  Enabled    Up    Up          XFI            10Gb/s  9600
Global   Global
1A-ETH-4  Bridge-1-1  Enabled    Up    Up          XFI            10Gb/s  9600
Global   Global
#
```

Related Information

- "disablesm Command" on page 24
- "enablesm Command" on page 30
- "Linux Shells for Hardware Commands" on page 3

`showioadapters` Command

Displays I/O adapters.

Syntax

```
showioadapters
```

Description

This hardware command displays the I/O adapter GUID to LID to gateway connector mapping. An I/O adapter (IOA) represents a HCA port with an EoIB driver, that is able to reach the relevant gateway instance.

Example

The following example shows how to display the I/O adapter GUID to LID to gateway connector mapping with the `showioadapters` command.

Note – The following example is a portion of the output and will differ from your fabric.

```
# showioadapters
IOA_GUID                 NODE       LID  #vADPT FLAGS GW
------------------------ ---------- ---- ------ ----- --------
00:03:BA:00:01:00:E3:71 mnm34-60   14   0      HD    1A-ETH-4
00:03:BA:00:01:00:E3:72 mnm34-60   15   0      HD    1A-ETH-4
00:21:28:00:01:3E:CE:9F mnm34-55   5    0      HD    1A-ETH-4
00:21:28:00:01:3E:CE:A0 mnm34-55   1c   0      HD    1A-ETH-4
00:02:C9:03:00:08:91:AB mnm34-54   12   0      HD    1A-ETH-4
00:02:C9:03:00:08:91:AC mnm34-54   13   0      HD    1A-ETH-4
00:03:BA:00:01:00:E3:71 mnm34-60   14   0      HD    1A-ETH-3
.
.
.
00:03:BA:00:01:00:E3:71 mnm34-60   14   0      HD    0A-ETH-3
00:03:BA:00:01:00:E3:72 mnm34-60   15   0      HD    0A-ETH-3
00:21:28:00:01:3E:CE:9F mnm34-55   5    0      HD    0A-ETH-3
00:21:28:00:01:3E:CE:A0 mnm34-55   1c   0      HD    0A-ETH-3
00:02:C9:03:00:08:91:AB mnm34-54   12   0      HD    0A-ETH-3
00:02:C9:03:00:08:91:AC mnm34-54   13   0      HD    0A-ETH-3
#
```

Related Information

- "showvlan Command" on page 76
- "showvnics Command" on page 78

showpsufru Command

Displays power supply FRU ID information.

Syntax

`showpsufru` *slot*

where *slot* is the power supply slot (`0` or `1`).

Description

This hardware command displays FRU ID information for the power supplies.

The `showpsufru` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to display the FRU ID information for power supply 0 with the `showpsufru` command.

```
# showpsufru 0
 Sun_SpecPartNo          :  885-1165-02
 UNIX_Timestamp32        :  Sun Jan  3 15:35:39 2010
 Sun_Part_Number         :  3002143
 Sun_Serial_No           :  BF15WA
 Vendor_ID_Code          :  02a2
 Initial_HW_Dash_Level   :  02
 PSU_Voltage 1           :  0x04b0 (1200)
 PSU_Current 1           :  0x186a (6250)
 PSU_Voltage 2           :  0x014a (330)
```

```
 PSU_Current 2           :  0x012c (300)
 Sun_Hazard_Class_Code   :  Y
 IPMI_Board_Manufacturer :  EMERSON
 IPMI_Board_Product_Name :  A237
 IPMI_Board_Serial_Number:  1357ZHO-0952BF15WA
 IPMI_Board_Part_Number  :  300-2143-02
#
```

Related Information

- "showfruinfo Command" on page 63
- "Linux Shells for Hardware Commands" on page 3

showsmlog Command

Displays the Subnet Manager log.

Syntax

```
showsmlog [-h]
```

Description

This hardware command displays the log entries for the Subnet Manager within the management controller. The output of the command is an invocation of `less /var/log/opensm.log`. Tapping the space bar displays the next screen of the log. Pressing the Q key quits. The `-h` option provides help.

The `showsmlog` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to display the Subnet Manager log with the `showsmlog` command.

Note – The output of the example is a very small portion and will differ from your output.

```
# showsmlog
Aug 13 15:58:22 679655 [B75F1B90] 0x01 -> __osm_mcmr_rcv_join_mgrp: ERR 1B11:
method = Subn
AdmSet, scope_state = 0x1, component mask = 0x0000000000010003, expected comp
mask = 0x0000
0000000130c7, MGID: ff12:e01b:2::22:2000 from port 0x0003ba000100e371 (mnm34-60
HCA-1)
Aug 13 15:58:22 692653 [B75F1B90] 0x01 -> __osm_mcmr_rcv_join_mgrp: ERR 1B11:
method = Subn
AdmSet, scope_state = 0x1, component mask = 0x0000000000010003, expected comp
mask = 0x0000
0000000130c7, MGID: ff12:e01b:3::22:b000 from port 0x00212800013ece9f (mnm34-55
HCA-1)
.
.
.
```

Related Information

- "setloghost Command" on page 58
- "Linux Shells for Hardware Commands" on page 3

showtemps Command

Displays gateway temperatures.

Syntax

```
showtemps
```

Description

This hardware command displays internal temperatures for the gateway.

The `showtemps` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to display gateway temperatures with the `showtemps` command.

```
# showtemps
Back temperature 29
Front temperature 29
SP temperature 47
Switch temperature 40, maxtemperature 42
Bridge-0 temperature 40, maxtemperature 41
Bridge-1 temperature 45, maxtemperature 46
All temperatures OK
#
```

Related Information

- "env_test Command" on page 34
- "Linux Shells for Hardware Commands" on page 3

`showtopology` Command

Displays the topology.

Syntax

```
showtopology [-h]
```

Description

This hardware command displays the InfiniBand fabric topology. The `showtopology` command displays the topology in a format different than the `ibnetdiscover` command.

The output of the `showtopology` command follows this basic format for each node:

```
device_type  device_name
panel_label  link_status-> connected_device_type  connected_device_name  connected_panel_label
and
panel_label  link_status-> connected_device_type  connected_device_name  connected_panel_label
.
.
for each subsequent port of that device_type
```

Where:

- *device_type* is the type of InfiniBand device. If the device is a Sun product, the *device_type* might be:
  - `SUNBQNEM48` – Sun Blade 6048 InfiniBand QDR Switched Network Express Module
  - `DEV26418_01P` – Single-port Sun Blade X6275 Server Module
  - `DEV26418_02P` – Dual-port Sun Blade X6275 Server Module
  - `SUNDCS36QDR` – Sun Datacenter InfiniBand Switch 36
  - `SUNDCS72QDR` – Sun Datacenter InfiniBand Switch 72
  - `SUNDCS648QDR` – Sun Datacenter InfiniBand Switch 648
  - `SUNIBQDRGW` – Sun Network QDR InfiniBand Gateway Switch
  - `I4_GENERIC` – An I4 switch ship
- *device_name* is the node description of the device, the hostname and number, or the modified GUID.
- *panel_label* is the name of the connector, hard wire link, or port as defined by the `ibnl` files. For example, `C-6A` is connector `6A`, `BX1` is BridgeX slice `1`, `A-SW` is switch chip `A`, and `P1` is port `1`.
- *link_status* is one of the following:
  - `-1x-2.5G` – single data rate (SDR)
  - `-2x-5G` – dual data rate (DDR)
  - `-4x-10G` – quad data rate (QDR)
- *connected_device_type* is the device type at the other end of the link.
- *connected_device_name* is the name of the device at the other end of the link.

- *connected_panel_label* is the name of the connector or port at the other end of the link.

For example:

```
SUNIBQDRGW mnm34-96.example.com
  BX1/P1 -4x-10G-> SUNIBQDRGW mnm34-96.example.com A-SW/P4
  BX3/P1 -4x-10G-> SUNIBQDRGW mnm34-96.example.com A-SW/P2
  A-SW/P2 -4x-10G-> SUNIBQDRGW mnm34-96.example.com BX3/P1
  A-SW/P4 -4x-10G-> SUNIBQDRGW mnm34-96.example.com BX1/P1
```

The `showtopology` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to display the InfiniBand fabric topology with the `showtopology` command.

Note – The output for your InfiniBand fabric will differ from that in the example.

```
# showtopology
SUNIBQDRGW mnm34-98.example.com Bridge0
  BX1/P1 -4x-10G-> I4_GENERIC mnm34-98 P4
      -4x-10G-> I4_GENERIC mnm34-98 P3
  BX3/P1 -4x-10G-> I4_GENERIC mnm34-98 P2
      -4x-10G-> I4_GENERIC mnm34-98 P1

DEV26428_02P mnm34-55_HCA-1
  P1 -4x-10G-> SUNIBQDRGW mnm34-97.example.com C-7A
  P2 -4x-10G-> I4_GENERIC mnm34-98 P33

SUNIBQDRGW mnm34-97
  A-SW/P2 -4x-10G-> SUNIBQDRGW mnm34-97.example.com BX3/P1
  A-SW/P4 -4x-10G-> SUNIBQDRGW mnm34-97.example.com BX1/P1
  C-0B -4x-10G-> I4_GENERIC mnm34-98 P21
  C-0A -4x-10G-> I4_GENERIC mnm34-98 P20
  C-1B -4x-10G-> I4_GENERIC mnm34-98 P19
  C-1A -4x-10G-> I4_GENERIC mnm34-98 P22
  C-2B -4x-10G-> I4_GENERIC mnm34-98 P24
  C-2A -4x-10G-> I4_GENERIC mnm34-98 P23
  C-8A -4x-10G-> DEV26428_02P mnm34-60_HCA-1 P1
  C-7A -4x-10G-> DEV26428_02P mnm34-55_HCA-1 P1
```

```
  C-6A -4x-10G-> DEV26428_02P mnm34-54_HCA-1 P1
  BX1/P1 -4x-10G-> SUNIBQDRGW mnm34-97.example.com A-SW/P4
  BX3/P1 -4x-10G-> SUNIBQDRGW mnm34-97.example.com A-SW/P2

I4_GENERIC mnm34-98
  P1 -4x-10G-> SUNIBQDRGW mnm34-98.example.com Bridge 0
  P2 -4x-10G-> SUNIBQDRGW mnm34-98.example.com Bridge 0 BX3/P1
  P3 -4x-10G-> SUNIBQDRGW mnm34-98.example.com Bridge 0
  P4 -4x-10G-> SUNIBQDRGW mnm34-98.example.com Bridge 0 BX1/P1
  P19 -4x-10G-> SUNIBQDRGW mnm34-97.example.com C-1B
  P20 -4x-10G-> SUNIBQDRGW mnm34-97.example.com C-0A
  P21 -4x-10G-> SUNIBQDRGW mnm34-97.example.com C-0B
  P22 -4x-10G-> SUNIBQDRGW mnm34-97.example.com C-1A
  P23 -4x-10G-> SUNIBQDRGW mnm34-97.example.com C-2A
  P24 -4x-10G-> SUNIBQDRGW mnm34-97.example.com C-2B
  P31 -4x-10G-> DEV26428_02P mnm34-60_HCA-1 P2
  P33 -4x-10G-> DEV26428_02P mnm34-55_HCA-1 P2
  P35 -4x-10G-> DEV26428_02P mnm34-54_HCA-1 P2

DEV26428_02P mnm34-54_HCA-1
  P1 -4x-10G-> SUNIBQDRGW mnm34-97.example.com C-6A
  P2 -4x-10G-> I4_GENERIC mnm34-98 P35

DEV26428_02P mnm34-60_HCA-1
  P1 -4x-10G-> SUNIBQDRGW mnm34-97.example.com C-8A
  P2 -4x-10G-> I4_GENERIC mnm34-98 P31

# Created from mnm34-98 at Thu Aug 19 13:32:25 PDT 2010
#
```

Related Information

- "generatetopology Command" on page 37
- "matchtopology Command" on page 49
- "Linux Shells for Hardware Commands" on page 3

`showunhealthy` Command

Displays problematic components.

Syntax

```
showunhealthy
```

Description

This hardware command shows a list of gateway components that appear to have a problem. Unlike the `env_test` command, the `showunhealty` command *only* displays messages for components that have failed testing.

The `showunhealthy` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to display a list of unhealthy components in the gateway with the `showunhealthy` command.

```
# showunhealthy
OK - No unhealthy sensors
#
```

Related Information

- "env_test Command" on page 34
- "Linux Shells for Hardware Commands" on page 3

`showvlan` Command

Displays VLANs.

Syntax

```
showvlan connector [-h]
```

where *connector* is the name of the connector (`0A-ETH-1` to `0A-ETH-4`, `0A-ETH`, `1A-ETH-1` to `1A-ETH-4`, and `1A-ETH`).

Description

This hardware command displays the VLANs associated to the gateway connectors. If no connector is specified, all connectors are displayed. The `-h` option provides help.

The `showvlan` command is available from the `/SYS/Gateway_Mgmt` and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to display the VLANs associated with all connectors using the `showvlan` command.

Note – Your output will differ from that in the example.

```
# showvlan
 Connector/LAG  VLN  PKEY
 -------------  ---  ----
 0A-ETH-1        1   ffff
 0A-ETH-1        0   ffff
 0A-ETH-2        1   ffff
 0A-ETH-2        0   ffff
 0A-ETH-3        1   ffff
 0A-ETH-3        0   ffff
 0A-ETH-4        1   ffff
 0A-ETH-4        0   ffff
 1A-ETH-1        1   ffff
 1A-ETH-1        0   ffff
 1A-ETH-2        1   ffff
 1A-ETH-2        0   ffff
 1A-ETH-3        1   ffff
 1A-ETH-3        0   ffff
 1A-ETH-4        1   ffff
 1A-ETH-4        0   ffff
#
```

Related Information

- "createvlan Command" on page 14
- "deletevlan Command" on page 19
- "Linux Shells for Hardware Commands" on page 3

`showvnics` Command

Displays VNIC information.

Syntax

```
showvnics
```

Description

This hardware command displays VNICs and their parameters. The list of VNICs is in ascending order, with priority in this sequence:

- Connector number
- GUID
- State
- MAC address

The `showvnics` command is available from the `/SYS/Gateway_Mgmt` and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to display the state of the VNICs with the `showvnic` command.

Note – Your output will differ from that in the example.

```
# showvnics
ID  STATE    FLG IOA_GUID              NODE      IID  MAC             VLN PKEY   GW
--- --------  --- ----------------------- ---------- ---- ----------------- ---
----   --------
 9 UP          N 00:03:BA:00:01:00:E3:71 mnm34-60.example.com 0000
02:02:02:02:02:07 NO  9002   0A-ETH-1
 8 UP          N 00:03:BA:00:01:00:E3:72 mnm34-60.example.com 0000
02:02:02:02:02:06 NO  9002   0A-ETH-1
 4 UP          N 00:03:BA:00:01:00:E3:71 mnm34-60.example.com 0000
02:02:02:02:02:03 NO  ffff   0A-ETH-2
 5 UP          N 00:03:BA:00:01:00:E3:71 mnm34-60.example.com 0002
02:02:02:02:02:04 NO  ffff   0A-ETH-2
 3 DISABLED    N 00:03:BA:00:01:00:E3:71 mnm34-60.example.com 0000
02:02:02:02:02:02 NO  ffff   0A-ETH-2
 6 UP          N 00:03:BA:00:01:00:E3:72 mnm34-60.example.com 0000
02:02:02:02:02:05 NO  ffff   0A-ETH-2
 7 UP          N 00:03:BA:00:01:00:E3:71 mnm34-60.example.com 0000
02:02:02:02:02:01 NO  ffff   0A-ETH-3
#
```

Related Information

- "createvnic Command" on page 15
- "deletevnic Command" on page 20
- "disablevnic Command" on page 27
- "enablevnic Command" on page 33
- "Linux Shells for Hardware Commands" on page 3

smconfigtest Command

Tests Subnet Manager configuration.

Syntax

```
smconfigtest [-l][-h]
```

Description

This hardware command tests the Subnet Manager configuration and reports if it is corrupt.

The `smconfigtest` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `smconfigtest` command and their purposes:

Option	Purpose
`-l`	Does not output to syslog.
`-h`	Provides help.

Example

The following example shows how to test the Subnet Manager configuration with the `smconfigtest` command.

```
# smconfigtest
#
```

Related Information

- "getmaster Command" on page 39
- "Linux Shells for Hardware Commands" on page 3

`version` Command (Hardware)

Displays gateway version.

Syntax

```
version
```

Description

This hardware command shows the hardware and software versions, and date information for the gateway and management controller.

The `version` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Example

The following example shows how to display the version information with the `version` command.

```
# version
SUN DCS gw version: 1.3.2-1
Build time: Feb 17 2011 10:02:40
FPGA version: 0x33
SP board info:
Manufacturing Date: 2009.06.24
Serial Number: "NCD3R0527"
Hardware Revision: 0x0006
Firmware Revision: 0x0102
BIOS version: NOW1R112
BIOS date: 04/24/2009
#
```

Related Information

- *Gateway Administration*, displaying firmware version
- "Linux Shells for Hardware Commands" on page 3

Understanding InfiniBand Commands

The InfiniBand commands act upon or monitor many aspects of the InfiniBand fabric. A portion of the commands are available through the Oracle ILOM CLI interface with the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` targets. See "Linux Shells for InfiniBand Commands" on page 85.

With the standard Linux shell CLI interface, only the `root` user of the management controller can run *all* of the InfiniBand commands. The format of the InfiniBand commands is typically as follows:

`#` *command* `[`*option*`]` `[`*option*`]` `...`

Command Syntax	Links
`ibaddr [-d][-D][-G][-l][-g][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`][-V][-h] [`*lid*`|`*dr_path*`|`*guid*`]`	"ibaddr Command" on page 86
`ibcheckerrors [-h][-b][-v][-N][`*topology*`|-C` *ca_name* `-P` *ca_port* `-t` *timeout*`]`	"ibcheckerrors Command" on page 88
`ibchecknet [-h][-N][`*topology*`|-C` *ca_name* `-P` *ca_port* `-t` *timeout*`]`	"ibchecknet Command" on page 90
`ibchecknode [-h][-v][-N][-G][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`] [`*lid*`|`*dr_path*`|`*guid*`]`	"ibchecknode Command" on page 92
`ibcheckport [-h][-v][-N][-G][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]` *lid*`|`*guid* *port*	"ibcheckport Command" on page 93
`ibcheckportstate [-G][-h][-N][-v][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`][`*lid*`|`*guid*`]` *port*	"ibcheckportstate Command" on page 95
`ibcheckportwidth [-G][-h][-N][-v][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`][`*lid*`|`*guid*`]` *port*	"ibcheckportwidth Command" on page 97
`ibcheckstate [-h][-N][-v][`*topology*`|-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]`	"ibcheckstate Command" on page 98
`ibcheckwidth [-h][-N][-v][`*topology*`|-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]`	"ibcheckwidth Command" on page 100

Command Syntax	Links
`ibclearcounters [-h][topology|-C ca_name][-P ca_port][-t timeout]`	"ibclearcounters Command" on page 102
`ibclearerrors [-h][-N][topology|-C ca_name][-P ca_port][-t timeout]`	"ibclearerrors Command" on page 103
`ibdatacounters [-b][-h][-N][-v][topology|-C ca_name][-P ca_port][-t timeout]`	"ibdatacounters Command" on page 105
`ibdatacounts [-b][-G][-h][-N][-v][-C ca_name][-P ca_port][-t timeout] lid|guid port`	"ibdatacounts Command" on page 106
`ibdiagnet [-c count][-v][-r][-o outputdir][-t topology][-s system][-i device][-p port][-wt topology][-pm][-pc][-P PM = value][-lw 1x|4x|12x][-ls 2.5|5|10][-skip checks][-load_db file][-h][-V]`	"ibdiagnet Command" on page 108
`ibdiagpath -n[src_name,]dst_name|-l[src_lid,]dst_lid|-d p1,p2,p3,...[-c count][-v][-o outputdir][-t topology][-s system][-i device][-p port][-wt topology][-pm][-pc][-P PM = value][-lw 1x|4x|12x][-ls 2.5|5|10][-skip checks][-load_db file][-h][-V]`	"ibdiagpath Command" on page 113
`ibhosts [-h][topology|-C ca_name][-P ca_port][-t timeout]`	"ibhosts Command" on page 116
`ibnetdiscover [-d][-e][-v][-s][-l][-g][-H][-S][-R][-C ca_name][-P ca_port][-t timeout][-V][-p][-h][topology]`	"ibnetdiscover Command" on page 118
`ibnetstatus [-h]`	"ibnetstatus Command" on page 121
`ibnodes [-h][topology|-C ca_name][-P ca_port][-t timeout]`	"ibnodes Command" on page 123
`ibportstate [-d][-D][-e][-G][-h][-s smlid][-v][-C ca_name][-P ca_port][-t timeout] lid|dr_path|guid port [op]`	"ibportstate Command" on page 125
`ibroute [-d][-a][-n][-D][-e][-G][-h][-M][-s smlid][-v][-V][-C ca_name][-P ca_port][-t timeout][lid|dr_path|guid [startlid [endlid]]]`	"ibroute Command" on page 128
`ibrouters [-h][topology|-C ca_name][-P ca_port][-t timeout]`	"ibrouters Command" on page 130
`ibstat [-d][-e][-h][-l][-s][-p][-v][-V] ca_name [ca_port]`	"ibstat Command" on page 131
`ibstatus [-h][devname[:ib_port]]...`	"ibstatus Command" on page 133
`ibswitches [-h][topology|-C ca_name][-P ca_port][-t timeout]`	"ibswitches Command" on page 135
`ibsysstat [-d][-e][-G][-h][-s smlid][-v][-V][-C ca_name][-P ca_port][-t timeout][-o oui][-S] lid|guid [op]`	"ibsysstat Command" on page 136

Command Syntax	Links
`ibtracert [-d][-D][-G][-h][-m` *mlid*`][-s` *smlid*`][-v][-V][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`][`*lid*`|`*dr_path*`|`*guid* `[`*startlid* `[`*endlid*`]]]`	"ibtracert Command" on page 139
`perfquery [-d][-e][-G][-h][-a][-l][-r][-R][-v][-V][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`][`*lid*`|`*guid* `[[`*port*`][`*reset_mask*`]]`	"perfquery Command" on page 141
`saquery [-h][-d][-p][-N][-D][-S][-I][-L][-l][-G][-O][-U][-c][-s][-g][-m][-x][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`][--src-to-dst` *source*`:`*destination*`][--sgid-to-dgid` *source-destination*`][`*name*`|`*lid*`|`*guid*`]`	"saquery Command" on page 143
`sminfo [-d][-e] -s` *state* `-p` *priority* `-a` *activity* `[-D][-G][-h][-v][-V][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]` *smlid*`|`*smdr_path*	"sminfo Command" on page 146
`smpdump [-s][-D][-h][-V][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]` *lid*`|`*dr_path* *attr* `[`*mod*`]`	"smpdump Command" on page 148
`smpquery [-d][-D][-e][-G][-h][-v][-V][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]` *lid*`|`*dr_path*`|`*guid* `[`*op* *params*`]`	"smpquery Command" on page 150

Related Information

- "Understanding Hardware Commands" on page 1
- "Understanding SNMP MIB OIDs" on page 153

Linux Shells for InfiniBand Commands

The preferred method of accessing the Linux shell is through the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Using the `show` command on the `/SYS/Switch_Diag` target opens a restricted Linux shell that enables the `ilom-admin` user, `ilom-operator` user, and users with similar permissions to run diagnostic commands.

Using the `show` command on the `/SYS/Gateway_Mgmt` target opens a different restricted Linux shell that enables the `ilom-admin` user and users with similar permissions to run both diagnostic and gateway management commands.

Using the `show` command on the `/SYS/Fabric_Mgmt` target opens still an other restricted Linux shell that enables the `ilom-admin` user and users with similar permissions to run diagnostic, gateway, and fabric management commands.

Note – The `ilom-operator` user cannot access the Linux shell from either the `/SYS/Gateway_Mgmt` or /SYS/Fabric_Mgmt target.

The following table lists the InfiniBand commands and their availability from the respective Linux shell targets. Typing the `help all` command from within the restricted shells lists the commands available to that shell.

Note – InfiniBand commands that are not listed are unavailable from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, or `/SYS/Fabric_Mgmt` targets.

Command	/SYS/Switch_Diag	/SYS/Gateway_Mgmt	/SYS/Fabric_Mgmt
ibdiagnet			Available
ibhosts	Available	Available	Available
ibnetstatus	Available	Available	Available
ibnodes	Available	Available	Available
ibportstate	Available	Available	Available
ibroute	Available	Available	Available
ibrouters	Available	Available	Available
ibstat	Available	Available	Available
ibswitches	Available	Available	Available
ibtracert	Available	Available	Available
perfquery	Available	Available	Available
saquery			Available
smpquery	Available	Available	Available

Related Information

- "Linux Shells for Hardware Commands" on page 3

ibaddr Command

Queries InfiniBand addresses.

Syntax

`ibaddr [-d][-D][-G][-l][-g][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`][-V][-h] [`*lid*`|`*dr_path*`|`*guid*`]`

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *dr_path* is the directed path.
- *guid* is the global unique identifier.

Description

This InfiniBand software command displays the LID and range as well as the GID address of the port specified. The local port information is provided by default.

Note – This command is also used as a simple address resolver.

Options

The following table describes the options to the `ibaddr` command and their purposes:

Option	Purpose
`-d`	Sets the debug level. Can be used several times to increase the debug level.
`-D`	Uses the directed path address. The path is a comma-delimited sequence of out ports.
`-G`	Shows the LID range and GID for port GUID addresses.
`-l`	Shows the LID range only.
`-g`	Shows the GID address only.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Option	Purpose
-V	Displays the version information.
-h	Provides help.

Example

The following example shows how to display the local port's GID and LID range with the `ibaddr` command.

```
# ibaddr
GID fe80::21:283a:8389:a0a0 LID start 0xf end 0xf
#
```

Related Information

- `ibaddr` man page
- "ibroute Command" on page 128
- "ibtracert Command" on page 139

`ibcheckerrors` Command

Validates InfiniBand fabric and report errors.

Syntax

`ibcheckerrors [-h][-b][-v][-N][`*topology*`|-C` *ca_name* `-P` *ca_port* `-t` *timeout*`]`

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that uses the topology file created by the `ibnetdiscover` command to scan the InfiniBand fabric to validate the connectivity and report errors from the port counters.

Options

The following table describes the options to the `ibcheckerrors` command and their purposes:

Option	Purpose
`-h`	Provides help.
`-b`	Enables brief mode. Reduced output is only if errors are present. Does not identify the errors.
`-v`	Provides verbose output.
`-N`	Uses mono mode instead of color mode.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to check error counters for all LIDs in the InfiniBand fabric with the `ibcheckerrors` command.

Note – The output in the example is a portion of the full output.

```
# ibcheckerrors
#warn: counter SymbolErrors = 3121      (threshold 10) lid 25 port 255
#warn: counter RcvSwRelayErrors = 48545         (threshold 100) lid 25 port 255
#warn: counter XmtDiscards = 9789       (threshold 100) lid 25 port 255
Error check on lid 25 (Sun DCS 72 QDR FC switch nm2-72p-2) port all:  FAILED
#warn: counter RcvSwRelayErrors = 56839         (threshold 100) lid 25 port 28
Error check on lid 25 (Sun DCS 72 QDR FC switch nm2-72p-2) port 28:  FAILED
#warn: counter RcvSwRelayErrors = 56839         (threshold 100) lid 25 port 9
Error check on lid 25 (Sun DCS 72 QDR FC switch nm2-72p-2) port 9:  FAILED
```

```
#warn: counter XmtDiscards = 9714       (threshold 100) lid 25 port 1
Error check on lid 25 (Sun DCS 72 QDR FC switch nm2-72p-2) port 1:  FAILED
.
.
.
## Summary: 6 nodes checked, 0 bad nodes found
##          142 ports checked, 3 ports have errors beyond threshold
#
```

Related Information

- `ibcheckerrors` man page
- "ibchecknode Command" on page 92
- "ibcheckport Command" on page 93
- "ibnetdiscover Command" on page 118

`ibchecknet` Command

A simplified version of the `ibcheckerrors` command.

Syntax

`ibchecknet [-h][-N][`*topology*`|-C` *ca_name* `-P` *ca_port* `-t` *timeout*`]`

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that uses the topology file created by the `ibnetdiscover` command to scan the InfiniBand fabric to validate the connectivity and report errors from the port counters.

Options

The following table describes the options to the `ibchecknet` command and their purposes:

Option	Purpose
`-h`	Provides help.
`-N`	Uses mono mode instead of color mode.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to check error counters for all LIDs in the InfiniBand fabric with the `ibchecknet` command.

Note – The output in the example is a portion of the full output.

```
# ibchecknet
#warn: counter SymbolErrors = 3121      (threshold 10) lid 25 port 255
#warn: counter RcvSwRelayErrors = 48545         (threshold 100) lid 25 port 255
#warn: counter XmtDiscards = 9789       (threshold 100) lid 25 port 255
Error check on lid 25 (Sun DCS 72 QDR FC switch nm2-72p-2) port all:  FAILED
#warn: counter RcvSwRelayErrors = 56839         (threshold 100) lid 25 port 28
Error check on lid 25 (Sun DCS 72 QDR FC switch nm2-72p-2) port 28:  FAILED
#warn: counter RcvSwRelayErrors = 56839         (threshold 100) lid 25 port 9
Error check on lid 25 (Sun DCS 72 QDR FC switch nm2-72p-2) port 9:  FAILED
#warn: counter XmtDiscards = 9714       (threshold 100) lid 25 port 1
Error check on lid 25 (Sun DCS 72 QDR FC switch nm2-72p-2) port 1:  FAILED
.
.
.
## Summary: 6 nodes checked, 0 bad nodes found
##          142 ports checked, 0 bad ports found
##          3 ports have errors beyond threshold
#
```

Related Information

- `ibchecknet` man page
- "ibcheckerrors Command" on page 88
- "ibchecknode Command" on page 92
- "ibcheckport Command" on page 93
- "ibnetdiscover Command" on page 118

`ibchecknode` Command

Validates InfiniBand nodes and reports errors.

Syntax

`ibchecknode -v [-h][-N][-G][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]` `[`*lid*`|`*dr_path*`|`*guid*`]`

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *guid* is the global unique identifier.

Description

This InfiniBand command checks node connectivity and performs a simple check to verify the functionality of the specified node.

Note – The port address is a LID, unless the `-G` option is used to specify a GUID address.

Options

The following table describes the options to the `ibchecknode` command and their purposes:

Option	Purpose
`-h`	Provides help.
`-N`	Uses mono mode instead of color mode.
`-G`	Uses the port GUID address.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to check if LID 15 is active with the `ibchecknode` command.

```
# ibchecknode -v 15
Node check lid 15:  OK
#
```

Related Information

- `ibchecknode` man page
- "ibaddr Command" on page 86
- "smpquery Command" on page 150

`ibcheckport` Command

Validates InfiniBand ports and reports errors.

Syntax

`ibcheckport [-h][-v][-N][-G][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]` *lid*|*guid* *port*

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *guid* is the global unique identifier.
- *port* is the port being validated.

Description

This InfiniBand command checks port connectivity and performs simple sanity checks for the specified port.

Note – The port address is a LID, unless the `-G` option is used to specify a GUID address.

Options

The following table describes the options to the `ibcheckport` command and their purposes:

Option	Purpose
`-h`	Provides help.
`-v`	Provides verbose output.
`-N`	Uses mono mode instead of color mode.
`-G`	Uses the port GUID address.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to check port 23 on LID 15 with the `ibcheckport` command.

```
# ibcheckport -v 15 23
Port check lid 15 port 23:  OK
#
```

Related Information

- `ibcheckport` man page
- "getportstatus Command" on page 43
- "ibaddr Command" on page 86
- "smpquery Command" on page 150

`ibcheckportstate` Command

Validates an InfiniBand port.

Syntax

`ibcheckportstate -v [-G][-h][-N][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`][`*lid*`|`*guid*`] port`

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *guid* is the global unique identifier.
- *port* is the port being validated.

Description

This InfiniBand command checks the specified port for the logical (`Active`) and physical (`LinkUp`) states.

Note – The port address is a LID, unless the `-G` option is used to specify a GUID address.

Options

The following table describes the options to the `ibcheckportstate` command and their purposes:

Option	Purpose
`-G`	Uses the port GUID address.
`-h`	Provides help.
`-N`	Uses mono mode instead of color mode.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to check port 23 on LID 15 with the `ibcheckportstate` command.

```
# ibcheckportstate -v 15 23
Port check lid 15 port 23:  OK
#
```

Related Information

- `ibcheckportstate` man page
- "getportstatus Command" on page 43
- "ibaddr Command" on page 86
- "smpquery Command" on page 150

`ibcheckportwidth` Command

Validates InfiniBand ports for 1x link width.

Syntax

`ibcheckportwidth -v [-G][-h][-N][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`][`*lid*`|`*guid*`] port`

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *guid* is the global unique identifier.
- *port* is the port being validated.

Description

This InfiniBand command checks connectivity and if the specified port is at 1x link width.

Note – The port address is a LID, unless the `-G` option is used to specify a GUID address.

Options

The following table describes the options to the `ibcheckportwidth` command and their purposes:

Option	Purpose
`-G`	Uses the port GUID address.
`-h`	Provides help.
`-N`	Uses mono mode instead of color mode.

Option	Purpose
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to check the width of port 23 on LID 15 with the `ibcheckportwidth` command.

```
# ibcheckportwidth -v 15 23
Port check lid 15 port 23:  OK
#
```

Related Information

- `ibcheckportwidth` man page
- "getportstatus Command" on page 43
- "ibaddr Command" on page 86
- "smpquery Command" on page 150

`ibcheckstate` Command

Displays ports that are `LinkUp` but not `Active`.

Syntax

`ibcheckstate [-h][-N][-v][`*topology*`|-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]`

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that uses the topology file created by the `ibnetdiscover` command. The script scans the InfiniBand fabric to validate the port logical and physical states, and reports any ports that have a logical state other than `Active` or a physical state other than `LinkUp`.

Options

The following table describes the options to the `ibcheckstate` command and their purposes:

Option	Purpose
`-h`	Provides help.
`-N`	Uses mono mode instead of color mode.
`-v`	Provides verbose output.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to check the state of all ports with the `ibcheckstate` command.

Note – The output for your InfiniBand fabric will differ from that in the example.

```
# ibcheckstate -v
# Checking Switch: nodeguid 0x0021283a8389a0a0
Node check lid 15:  OK
Port check lid 15 port 23:  OK
Port check lid 15 port 19:  OK
.
.
.
# Checking Ca: nodeguid 0x0003ba000100e388
Node check lid 14:  OK
Port check lid 14 port 2:  OK
```

```
## Summary: 5 nodes checked, 0 bad nodes found
##          10 ports checked, 0 ports with bad state found
#
```

Related Information

- `ibcheckstate` man page
- "ibchecknode Command" on page 92
- "ibcheckportstate Command" on page 95
- "ibnetdiscover Command" on page 118

`ibcheckwidth` Command

Finds 1x links in the InfiniBand fabric.

Syntax

`ibcheckwidth [-h][-N][-v][`*topology*`|-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]`

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that uses the topology file created by the `ibnetdiscover` command. The script scans the InfiniBand fabric to validate `Active` link widths and report the links that are 1x links.

Options

The following table describes the options to the `ibcheckwidth` command and their purposes:

Option	Purpose
`-h`	Provides help.
`-N`	Uses mono mode instead of color mode.
`-v`	Provides verbose output.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to check the 1x links for all ports with the `ibcheckwidth` command.

Note – The output for your InfiniBand fabric will differ from that in the example.

```
# ibcheckwidth -v
# Checking Switch: nodeguid 0x0021283a8389a0a0
Node check lid 15:  OK
Port check lid 15 port 23:  OK
Port check lid 15 port 19:  OK
Port check lid 15 port 17:  OK
.
.
.
# Checking Ca: nodeguid 0x0003ba000100e388
Node check lid 14:  OK
Port check lid 14 port 2:  OK

## Summary: 5 nodes checked, 0 bad nodes found
##          10 ports checked, 0 ports with 1x width in error found
#
```

Related Information

- `ibcheckwidth` man page

`ibclearcounters` Command

Clears port counters in the InfiniBand fabric.

Syntax

`ibclearcounters [-h][`*topology*`|-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]`

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that clears the Performance Manager agent port counters by either discovering the InfiniBand fabric topology or using an existing topology file. The counters are:

- `XmtData`
- `RcvData`
- `XmtPkts`
- `RcvPkts`

Options

The following table describes the options to the `ibclearcounters` command and their purposes:

Option	Purpose
`-h`	Provides help.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to clear the Performance Manager agent port counters with the `ibclearcounters` command.

```
# ibclearcounters
## Summary: 5 nodes cleared 0 errors
#
```

Related Information

- `ibclearcounters` man page
- "ibnetdiscover Command" on page 118
- "perfquery Command" on page 141

`ibclearerrors` Command

Clears error counters in the InfiniBand fabric.

Syntax

`ibclearerrors [-h][-N][`*topology*`|-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]`

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that clears the Performance Manager agent error counters in `PortCounters` by either discovering the InfiniBand fabric topology or using an existing topology file.

Options

The following table describes the options to the `ibclearerrors` command and their purposes:

Option	Purpose
`-h`	Provides help.
`-N`	Uses mono mode instead of color mode.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to clear all error counters with the `ibclearerrors` command.

```
# ibclearerrors
## Summary: 5 nodes cleared 0 errors
#
```

Related Information

- `ibclearerrors` man page
- "ibnetdiscover Command" on page 118

■ "perfquery Command" on page 141

ibdatacounters Command

Queries the InfiniBand fabric for data counters.

Syntax

`ibdatacounters [-b][-h][-N][-v][`*topology*`|-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]`

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that uses the topology file created by the `ibnetdiscover` command. The script scans the InfiniBand fabric to validate the connectivity and reports the values of the data counters.

Options

The following table describes the options to the `ibdatacounters` command and their purposes:

Option	Purpose
`-b`	Enables brief mode. Reduced output is only if errors are present. Does not identify the errors.
`-h`	Provides help.
`-N`	Uses mono mode instead of color mode.
`-v`	Provides verbose output.

Option	Purpose
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to display the data counters for all ports with the `ibdatacounters` command.

Note – You might see more output than what is in the example.

```
# ibdatacounters

## Summary: 5 nodes checked, 0 bad nodes found
##          10 ports checked
#
```

Related Information

- `ibdatacounters` man page
- "ibdatacounts Command" on page 106
- "ibnetdiscover Command" on page 118

`ibdatacounts` Command

Displays InfiniBand fabric port data counters.

Syntax

`ibdatacounts [-b][-G][-h][-N][-v][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]` *lid*|*guid* *port*

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *guid* is the global unique identifier.
- *port* is the port being validated.

Description

This InfiniBand command returns the Performance Manager agent data counters from a specified port or node.

Note – The port address is a LID, unless the `-G` option is used to specify a GUID address.

Options

The following table describes the options to the `ibdatacounts` command and their purposes:

Option	Purpose
`-b`	Enables brief mode. Reduced output is only if errors are present. Does not identify the errors.
`-G`	Uses the port GUID address.
`-h`	Provides help.
`-N`	Uses mono mode instead of color mode.
`-v`	Provides verbose output.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to display the data counters for LID 15, port 23 with the `ibdatacounts` command.

```
# ibdatacounts 15 23
# Port counters: Lid 15 port 23
XmtData:.......................6048
RcvData:.......................6048
XmtPkts:.......................84
RcvPkts:.......................84
#
```

Related Information

- `ibdatacounts` man page
- "ibaddr Command" on page 86
- "perfquery Command" on page 141

`ibdiagnet` Command

Performs InfiniBand fabric diagnostic.

Syntax

`ibdiagnet [-c` *count*`][-v][-r][-o` *outputdir*`][-t` *topology*`][-s` *system*`][-i` *device*`][-p` *port*`][-wt` *topology*`][-pm][-pc][-P` *PM* `=` *value*`][-lw 1x|4x|12x][-ls 2.5|5|10][-skip` *checks*`][-load_db` *file*`][-h][-V]`

where:

- *count* is the number of packets.
- *outputdir* is the output directory.
- *topology* is the topology file.
- *system* is the local system name.
- *device* is the index of the device connecting to the InfiniBand fabric.
- *port* is the port of the device.
- *PM* is the Performance Manager counter number.

- *value* is the threshold of the Performance Manager counter.
- *checks* is one or more strings that identify the checks made:
  - `dup_guids`
  - `zero_guids`
  - `pm`
  - `logical_state`
  - `part`
  - `ipoib`
  - `all`
- *file* is the subnet database `.db` file.

Description

This InfiniBand command scans the InfiniBand fabric using directed route packets, extracting all the available information regarding the connectivity and devices. This command produces a set of files in the output directory. By default, the output directory is `/tmp`. The following table describes the files.

File Name	Description
`ibdiagnet.log`	Dump of all the application reports generated according to the provided flags.
`ibdiagnet.lst`	List of all the nodes, ports, and links in the fabric.
`ibdiagnet.fdbs`	Dump of the unicast forwarding tables of the fabric switches.
`ibdiagnet.mcfdbs`	Dump of the multicast forwarding tables of the fabric switches.
`ibdiagnet.masks`	In case of duplicate port/node GUIDs, this file includes the map between masked GUIDs and real GUIDs.
`ibdiagnet.sm`	List of all the Subnet Manager (state and priority) in the fabric.
`ibdiagnet.pm`	Dump of the Performance Manager counters values, for the fabric links.
`ibdiagnet.pkey`	Dump of the existing partitions and their member host ports.
`ibdiagnet.mcg`	Dump of the multicast groups, their properties, and member host ports.
`ibdiagnet.db`	Dump of the internal subnet database. You can load this file in later runs using the `-load_db` option.

During the discovery phase, the command also checks for duplicate node/port GUIDs in the InfiniBand fabric. If such an error is detected, it is displayed on the standard output.

After the discovery phase is completed, directed route packets are sent multiple times to detect possible problematic paths on which packets might be lost. A report of suspected bad links is displayed on the standard output.

If requested with the `-r` option, a full report of fabric qualities is displayed, including:

- Subnet Manager report
- Number of nodes and systems
- Hop-count information containing maximal hop-count, an example path, and a hop-count histogram
- All CA-to-CA paths traced
- Credit loop report
- MGID-MLID-HCAs multicast group and report
- Partitions report
- IPoIB report

Note – If the InfiniBand fabric includes only one CA, then CA-to-CA paths are not reported. Additionally, if a topology file is provided, the `ibdiagnet` command uses the names defined in the topology file for the output reports.

The `ibdiagnet` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `ibdiagnet` command and their purposes:

Option	Purpose
`-c`	Sets the minimum number of packets sent across each link.
`-v`	Provides verbose output.
`-r`	Provides a report of fabric qualities.
`-t`	Specifies the topology file name.
`-s`	Specifies the local system name.
`-i`	In the case of multiple devices on the local system, this option specifies the index of the device of the port used to connect to the InfiniBand fabric.
`-p`	Specifies the local device port number used to connect to the InfiniBand fabric.

Option	Purpose
-o	Specifies the output directory.
-lw	Specifies the expected link width.
-ls	Specifies the expected link speed.
-pm	Dumps all the fabric link Performance Manager counters into ibdiagnet.pm.
-pc	Resets all the fabric link Performance Manager counters.
-P	Uses the Performance Manager counter of PM set to the threshold of value.
-skip	Skips the executions of the selected checks. One or more checks can be specified.
-wt	Writes out the discovered topology into the given file.
-load_db	Loads subnet data from the given .db file and skips the subnet discovery stage. **Note -** Some checks require actual subnet discovery and are disabled if load_db is specified. Those checks are for duplicate/zero GUIDs, link state, and Subnet Manager status.
-h	Provides help.
-V	Displays the version information.

Example

The following example shows how to test the InfiniBand fabric with the ibdiagnet command. The command checks for 4x link width and 10 Gbyte/sec speed, dumps the Performance Manager counters, and then clears the counters.

Note – The output for your InfiniBand fabric will differ from that in the example.

```
# ibdiagnet -lw 4x -ls 10 -pm -pc -skip dup_guids
Loading IBDIAGNET from: /usr/lib/ibdiagnet1.2
-W- Topology file is not specified.
  Reports regarding cluster links will use direct routes.
Loading IBDM from: /usr/lib/ibdm1.2
-I- Using port 0 as the local port.
-I- Discovering ... 2 nodes (1 Switches & 1 CA-s) discovered.
-I---------------------------------------------------
-I- Bad Guids/LIDs Info
-I---------------------------------------------------
-I- No bad Guids were found
-I---------------------------------------------------
```

```
-I- Links With Logical State = INIT
-I----------------------------------------------------
-I- No bad Links (with logical state = INIT) were found
-I----------------------------------------------------
-I- PM Counters Info
-I----------------------------------------------------
-I- No illegal PM counters values were found
-I----------------------------------------------------
-I- Links With links width != 4x (as set by -lw option)
-I----------------------------------------------------
-I- No unmatched Links (with width != 4x) were found
-I----------------------------------------------------
-I- Links With links speed != 10 (as set by -ls option)
-I----------------------------------------------------
-I- No unmatched Links (with speed != 10) were found
-I----------------------------------------------------
-I- Fabric Partitions Report (see ibdiagnet.pkey for a full hosts list)
-I----------------------------------------------------
-I----------------------------------------------------
-I- IPoIB Subnets Check
-I----------------------------------------------------
-I- Subnet: IPv4 PKey:0x7fff QKey:0x00000b1b MTU:2048Byte rate:10Gbps SL:0x00
-W- No members found for group
-I----------------------------------------------------
-I- Bad Links Info
-I- No bad link were found
-I----------------------------------------------------
------------------------------------------------------------------
-I- Stages Status Report:
  STAGE                                 Errors Warnings
  Bad GUIDs/LIDs Check                  0      0
  Link State Active Check               0      0
  Performance Counters Report           0      0
  Specific Link Width Check             0      0
  Specific Link Speed Check             0      0
  Partitions Check                      0      0
  IPoIB Subnets Check                   0      1
Please see /tmp/ibdiagnet.log for complete log
------------------------------------------------------------------
-I- Done. Run time was 2 seconds.
#
```

Related Information

- ibdiagnet man page
- "ibdiagpath Command" on page 113
- "Linux Shells for InfiniBand Commands" on page 85

`ibdiagpath` Command

Traces the InfiniBand fabric diagnostic path.

Syntax

`ibdiagpath -n[`*src_name*`,]`*dst_name*`|-l[`*src_lid*`,]`*dst_lid*`|-d` *p1,p2,p3,...*`[-c` *count*`][-v][-o` *outputdir*`][-t` *topology*`][-s` *system*`][-i` *device*`][-p` *port*`][-wt` *topology*`][-pm][-pc][-P` *PM* `=` *value*`][-lw 1x|4x|12x][-ls 2.5|5|10][-h][-V]`

where:

- *src_name* is the source port.
- *dst_name* is the destination port.
- *src_lid* is the source LID.
- *dst_lid* is the destination LID.
- *p1,p2,p3,...* is the directed route.
- *count* is the number of packets.
- *outputdir* is the output directory.
- *topology* is the topology file.
- *system* is the local system name.
- *device* is the index of the device connecting to the InfiniBand fabric.
- *port* is the port of the device.
- *PM* is the Performance Manager counter number.
- *value* is the threshold of the Performance Manager counter.

Description

This InfiniBand command traces a path between two endpoints and provides information regarding the nodes and ports traversed along the path. The command uses device-specific health queries for the different devices encountered. The way the `ibdiagpath` command operates is determined from the addressing mode specified on the command line:

- If directed route addressing is used, the local node is the source node and the route to the destination port is known.

- If LID route addressing is used, the source and destination ports of a route are specified by their LIDs.

In LID route addressing, the actual path from the local port to the source port, and from the source port to the destination port, is defined by means of Subnet Management Linear Forwarding Table queries of the switch nodes along those paths.

Note – When the `ibdiagpath` command queries the performance counters along the path between the source and destination ports, the command always traverses the LID route, regardless of whether a directed route is specified. If one or more links along the LID route are not in the `Active` state, the `ibdiagpath` command reports an error.

This command produces a set of files in the output directory. By default, the output directory is `/tmp`. The following table describes the files.

File Name	Description
`ibdiagpath.log`	Dump of all the application reports generated according to the provided flags.
`ibdiagpath.pm`	Dump of the Performance Manager counters values, for the fabric links.

Options

The following table describes the options to the `ibdiagpath` command and their purposes:

Option	Purpose
`-n`	Identifies the source and destination ports.
`-l`	Identifies the source and destination LIDs.
`-d`	Sets directed route from the local node to the destination node.
`-c`	Sets the minimum number of packets sent across each link.
`-v`	Provides verbose output.
`-t`	Specifies the topology file name.
`-s`	Specifies the local system name.
`-i`	In the case of multiple devices on the local system, this option specifies the index of the device of the port used to connect to the InfiniBand fabric.
`-p`	Specifies the local device port number used to connect to the InfiniBand fabric.
`-o`	Specifies the output directory.

Option	Purpose
-lw	Specifies the expected link width.
-ls	Specifies the expected link speed.
-pm	Dumps all the fabric link Performance Manager counters into ibdiagpath.pm.
-pc	Resets all the fabric link Performance Manager counters.
-P	Use the Performance Manager counter of PM set to the threshold of value.
-h	Provides help.
-V	Displays the version information.

Example

The following example shows how to display the route from LID 15 to LID 14 with the ibdiagpath command.

Note – The output for your InfiniBand fabric will differ from that in the example.

```
# ibdiagpath -l 14,15
Loading IBDIAGPATH from: /usr/lib/ibdiagpath1.2
-W- Topology file is not specified.
  Reports regarding cluster links will use direct routes.
Loading IBDM from: /usr/lib/ibdm1.2
-I- Using port 0 as the local port.
-I----------------------------------------------------
-I- Traversing the path from local to source
-I----------------------------------------------------
-I- From: lid=0x000f guid=0x0021283a8389a0a0 dev=48438 Port=23
-I- To:   lid=0x000e guid=0x0003ba000100e38a dev=26428 mnm33-43/P2
-I----------------------------------------------------
-I- Traversing the path from source to destination
-I----------------------------------------------------
-I- From: lid=0x000e guid=0x0003ba000100e38a dev=26428 mnm33-43/P2
-I- To:   lid=0x000f guid=0x0021283a8389a0a0 dev=48438 Port=23
-I----------------------------------------------------
-I- PM Counters Info
-I----------------------------------------------------
-I- No illegal PM counters values were found
-I----------------------------------------------------
-I- Path Partitions Report
-I----------------------------------------------------
-I- Source mnm33-43/P2 lid=0x000e guid=0x0003ba000100e38a dev=26428 Port 2
  PKeys:0xffff
```

```
-I- Destination  lid=0x000f guid=0x0021283a8389a0a0 dev=48438
  PKeys:0xffff
-I- Path shared PKeys: 0xffff
-I-------------------------------------------------
-I- IPoIB Path Check
-I-------------------------------------------------
-I- Subnet: IPv4 PKey:0x7fff QKey:0x00000b1b MTU:2048Byte rate:10Gbps SL:0x00
-I-------------------------------------------------
-I- QoS on Path Check
-I-------------------------------------------------
-I- The following SLs can be used:0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
---------------------------------------------------------------
-I- Stages Status Report:
  STAGE                                  Errors Warnings
  LFT Traversal: local to source         0      0
  LFT Traversal: source to destination   0      0
  Performance Counters Report            0      0
  Path Partitions Check                  0      0
  Path IPoIB Check                       0      0
  QoS on Path Check                      0      0
Please see /tmp/ibdiagpath.log for complete log
---------------------------------------------------------------
-I- Done. Run time was 0 seconds.
#
```

Related Information

- `ibdiagpath` man page
- "ibdiagnet Command" on page 108

`ibhosts` Command

Displays host nodes.

Syntax

`ibhosts [-h][`*topology*`|-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]`

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.

- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that discovers the InfiniBand fabric topology or uses the existing topology file to extract the channel adapter nodes.

The `ibhosts` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `ibhosts` command and their purposes:

Option	Purpose
`-h`	Provides help.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to display the host node GUIDs with the `ibhosts` command.

Note – The output in the example is a portion of the full output and varies for each InfiniBand topology.

```
# ibhosts
Ca      : 0x00212856cd22c000 ports 2 "SUN IB QDR GW switch mnm34-97.example.com
Bridge 0"
Ca      : 0x00212856cd22c040 ports 2 "SUN IB QDR GW switch mnm34-97.example.com
Bridge 1"
Ca      : 0x0002c903000891aa ports 2 "mnm34-54 HCA-1"
```

```
Ca      : 0x00212800013ece9e ports 2 "mnm34-55 HCA-1"
Ca      : 0x0003ba000100e370 ports 2 "mnm34-60 HCA-1"
.
.
.
#
```

Related Information

- `ibhosts` man page
- "ibnetdiscover Command" on page 118
- "ibnodes Command" on page 123
- "Linux Shells for InfiniBand Commands" on page 85

`ibnetdiscover` Command

Discovers the InfiniBand topology.

Syntax

`ibnetdiscover [-d][-e][-v][-s][-l][-g][-H][-S][-R][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`][-V][--node-name-map` *map*`][-p][-h][`*topology*`]`

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *map* is the file name of the node name map.
- *topology* is the topology file.

Description

This InfiniBand command performs InfiniBand fabric discovery and outputs a human readable topology file. Nodes, node types, node descriptions, links, port numbers, port LIDs, and GUIDs are displayed. The output is directed to a topology file, if the file name is specified.

The output of the topology file follows this basic format for each node:

```
vendid=vendor_ID_in_hex
devid=device _ID_in_hex
and
sysimgguid=GUID_in_hex
and/or
switchguid=GUID_in_hex(portGUID_in_hex)
Switch ports_total "type-nodeGUID_in_hex" # "NodeDescription" base port 0 lid LID lmc 0
or
caguid=GUID_in_hex
Ca ports_total "type-nodeGUID_in_hex" # "NodeDescription"
and
[port] "type-nodeGUID_in_hex"[port](portGUID_in_hex) # "NodeDescription" lid LID widthspeed
[port] "type-nodeGUID_in_hex"[port](portGUID_in_hex) # "NodeDescription" lid LID widthspeed
.
.
.
```

For example:

```
vendid=0x2c9
devid=0xbd36
sysimgguid=0x21283a8389a0a3
switchguid=0x21283a8389a0a0(21283a8389a0a0)
Switch  36 "S-0021283a8389a0a0" # "Sun DCS 36 QDR switch localhost" enhanced
port 0 lid 15 lmc 0
[23]    "H-0003ba000100e388"[2](3ba000100e38a) # "mnm33-43 HCA-1" lid 14 4xQDR
.
.
.
```

Options

The following table describes the options to the `ibnetdiscover` command and their purposes:

Option	Purpose
`-d`	Sets the debug level. Can be used several times to increase the debug level.
`-e`	Displays send and receive errors.
`-v`	Provides verbose output.

Option	Purpose
-s	Shows more information.
-l	Lists the connected nodes.
-g	Shows the grouping and switch external ports correspondence.
-H	Lists the connected channel adapters.
-S	Lists the connected switches.
-R	Lists the connected routers.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.
-V	Displays the version information.
--node-name-map	Reads the node name map file.
-p	Returns a list of connected ports, including status information: • LID • portnum • GUID • link width • link speed • NodeDescription
-h	Provides help.

Example

The following example shows how to discover the InfiniBand fabric topology with the `ibnetdiscover` command.

Note – The output for your InfiniBand fabric will differ from that in the example.

```
# ibnetdiscover
#
# Topology file: generated on Sat Apr 13 22:28:55 2002
#
# Max of 1 hops discovered
# Initiated from node 0021283a8389a0a0 port 0021283a8389a0a0
vendid=0x2c9
devid=0xbd36
sysimgguid=0x21283a8389a0a3
```

```
switchguid=0x21283a8389a0a0(21283a8389a0a0)
Switch  36 "S-0021283a8389a0a0" # "Sun DCS 36 QDR switch localhost" enhanced port
0 lid 15 lmc 0
[23]    "H-0003ba000100e388"[2](3ba000100e38a) # "mnm33-43 HCA-1" lid 14 4xQDR

vendid=0x2c9
devid=0x673c
sysimgguid=0x3ba000100e38b
caguid=0x3ba000100e388
Ca      2 "H-0003ba000100e388" # "mnm33-43 HCA-1"
[2](3ba000100e38a)      "S-0021283a8389a0a0"[23] # lid 14 lmc 0 "Sun DCS 36 QDR
switch localhost" lid 15 4xQDR
#
```

Related Information

- `ibnetdiscover` man page

`ibnetstatus` Command

Displays status of the InfiniBand fabric.

Syntax

```
ibnetstatus [-h]
```

Description

This InfiniBand command provides a short status report of the InfiniBand fabric. It provides output equivalent to `ibdiagnet -ls 10 -lw 4x`.

The `ibnetstatus` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The `-h` option provides help.

Example

The following example shows how to display the status of the InfiniBand fabric with the `ibnetstatus` command.

Note – The output for your InfiniBand fabric will differ from that in the example.

```
# ibnetstatus
Loading IBDIAGNET from: /usr/lib/ibdiagnet1.2
-W- Topology file is not specified.
  Reports regarding cluster links will use direct routes.
Loading IBDM from: /usr/lib/ibdm1.2
-I- Using port 0 as the local port.
-I- Discovering ... 9 nodes (2 Switches & 7 CA-s) discovered.
-I---------------------------------------------------
-I- Bad Guids/LIDs Info
-I---------------------------------------------------
-I- skip option set. no report will be issued
-I---------------------------------------------------
-I- Links With Logical State = INIT
-I---------------------------------------------------
-I- No bad Links (with logical state = INIT) were found
-I---------------------------------------------------
-I- PM Counters Info
-I---------------------------------------------------
-W- lid=0x0010 guid=0x00212856cfe2c0a0 dev=48438 Port=1
 Performance Monitor counter     : Value
 symbol_error_counter            : 0xffff (overflow)
-W- lid=0x0010 guid=0x00212856cfe2c0a0 dev=48438 Port=2
 Performance Monitor counter     : Value
 symbol_error_counter            : 0xffff (overflow)
-W- lid=0x0010 guid=0x00212856cfe2c0a0 dev=48438 Port=3
 Performance Monitor counter     : Value
 symbol_error_counter            : 0xffff (overflow)
-W- lid=0x0010 guid=0x00212856cfe2c0a0 dev=48438 Port=4
 Performance Monitor counter     : Value
 symbol_error_counter            : 0xffff (overflow)
-I---------------------------------------------------
-I- Links With links width != 4x (as set by -lw option)
-I---------------------------------------------------
```

```
-I- No unmatched Links (with width != 4x) were found
-I---------------------------------------------------
-I- Links With links speed != 10 (as set by -ls option)
-I---------------------------------------------------
-I- No unmatched Links (with speed != 10) were found
-I---------------------------------------------------
-I- Fabric Partitions Report (see ibdiagnet.pkey for a full hosts list)
-I---------------------------------------------------
-I---------------------------------------------------
-I- IPoIB Subnets Check
-I---------------------------------------------------
-I- Subnet: IPv4 PKey:0x0001 QKey:0x00000b1b MTU:2048Byte rate:10Gbps SL:0x00
-W- No members found for group
-I- Subnet: IPv4 PKey:0x7fff QKey:0x00000b1b MTU:2048Byte rate:10Gbps SL:0x00
-W- No members found for group
-I---------------------------------------------------
-I- Bad Links Info
-I- No bad link were found
-I---------------------------------------------------
-I- Stages Status Report:
 STAGE                                Errors Warnings
 Bad GUIDs/LIDs Check                 0      0
 Link State Active Check              0      0
 Performance Counters Report          0      4
 Specific Link Width Check            0      0
 Specific Link Speed Check            0      0
 Partitions Check                     0      0
 IPoIB Subnets Check                  0      2
-----------------------------------------------------------------
-I- Done. Run time was 22 seconds.
#
```

Related Information

- `ibnetstatus` man page
- "ibdiagnet Command" on page 108
- "Linux Shells for InfiniBand Commands" on page 85

`ibnodes` Command

Displays InfiniBand nodes in topology.

Syntax

`ibnodes [-h][`*topology*`|-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]`

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script that discovers the InfiniBand fabric topology or uses the existing topology file to extract the InfiniBand nodes of the channel adapters, switches, and routers.

The `ibnodes` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `ibnodes` command and their purposes:

Option	Purpose
`-h`	Provides help.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to display the node GUIDs with the `ibnodes` command.

Note – The output for your InfiniBand fabric will differ from that in the example.

```
# ibnodes
Ca      : 0x0003ba000100e388 ports 2 "mnm33-43 HCA-1"
Switch  : 0x0021283a8389a0a0 ports 36 "Sun DCS 36 QDR switch localhost" enhanced
port 0 lid 15 lmc 0
#
```

Related Information

- `ibnodes` man page
- "ibnetdiscover Command" on page 118
- "Linux Shells for InfiniBand Commands" on page 85

`ibportstate` Command

Manages the state and link speed of an InfiniBand port.

Syntax

`ibportstate [-d][-D][-e][-G][-h][-s` *smlid*`][-v][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]` *lid*|*dr_path*|*guid* *port* `[`*op*`]`

where:

- *smlid* is the Subnet Manager LID.
- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *dr_path* is the directed path.
- *guid* is the global unique identifier.
- *port* is the port being validated.
- *op* is the operation to perform on the port:

- `enable`
- `disable`
- `reset`
- `speed` *number* (where *number* is `1` for SDR, `2` for DDR, and `4` for QDR)
- `query` (default)

Description

This InfiniBand command queries the logical and physical state of an InfiniBand port. The command can return the link width and speed of a switch chip port, as well as enabling, disabling, or resetting the port. The command can also set the link speed of any InfiniBand port.

Note – State changes made with the `ibportstate` command are not recognized by the `disableswitchport`, `enableswitchport`, or `listlinkup` commands.

Note – Speed changes are not affected until the port undergoes link renegotiation. Additionally, speed values are additive for enabling. For example, `speed 7` is 2.5, 5.0, and 10.0 Gbyte/sec.

The `ibportstate` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `ibportstate` command and their purposes:

Option	Purpose
`-d`	Sets the debug level. Can be used several times to increase the debug level.
`-D`	Uses the directed path address. The path is a comma-delimited sequence of out ports.
`-e`	Displays send and receive errors.
`-G`	Uses the port GUID address.
`-h`	Provides help.

Option	Purpose
`-s`	Uses *smlid* as the target LID for Subnet Manager or Subnet Administrator queries.
`-v`	Provides verbose output.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to query the state and link speed of LID 15, port 23 with the `ibportstate` command.

```
# ibportstate 15 23
PortInfo:
# Port info: Lid 15 port 23
LinkState:.....................Active
PhysLinkState:.................LinkUp
LinkWidthSupported:............1X or 4X
LinkWidthEnabled:..............1X or 4X
LinkWidthActive:...............4X
LinkSpeedSupported:............2.5 Gbps or 5.0 Gbps or 10.0 Gbps
LinkSpeedEnabled:..............2.5 Gbps or 5.0 Gbps or 10.0 Gbps
LinkSpeedActive:...............10.0 Gbps
Peer PortInfo:
# Port info: Lid 15 DR path slid 15; dlid 65535; 0,23
LinkState:.....................Active
PhysLinkState:.................LinkUp
LinkWidthSupported:............1X or 4X
LinkWidthEnabled:..............1X or 4X
LinkWidthActive:...............4X
LinkSpeedSupported:............2.5 Gbps or 5.0 Gbps or 10.0 Gbps
LinkSpeedEnabled:..............2.5 Gbps or 5.0 Gbps or 10.0 Gbps
LinkSpeedActive:...............10.0 Gbps
#
```

Related Information

- `ibportstate` man page
- “disableswitchport Command” on page 25
- “enablesm Command” on page 30
- “ibdevreset Command” on page 46

- "listlinkup Command" on page 47
- "Linux Shells for InfiniBand Commands" on page 85

`ibroute` Command

Queries InfiniBand switch forwarding tables.

Syntax

`ibroute [-d][-a][-n][-D][-e][-G][-h][-M][-s` *smlid*`][-v][-V][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`][`*lid*`|`*dr_path*`|`*guid* `[`*startlid* `[`*endlid*`]]]`

where:

- *smlid* is the Subnet Manager LID.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *dr_path* is the directed path.
- *guid* is the global unique identifier.
- *startlid* is the starting local identifier.
- *endlid* is the ending local identifier.

Description

This InfiniBand command uses SMPs to display the forwarding tables for the specified switch LID and optionally, the LID range. By default, the range is all valid entries from `1` to `FDBTop`.

The `ibroute` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `ibroute` command and their purposes:

Option	Purpose
`-a`	Shows all LIDs in the range, including invalid entries.
`-n`	Does not try to resolve destinations.
`-d`	Sets the debug level. Can be used several times to increase the debug level.
`-D`	Uses the directed path address. The path is a comma-delimited sequence of out ports.
`-e`	Displays send and receive errors.
`-G`	Uses the port GUID address.
`-h`	Provides help.
`-M`	Shows multicast forwarding tables.
`-s`	Uses *smlid* as the target LID for Subnet Manager or Subnet Administrator queries.
`-v`	Provides verbose output.
`-V`	Displays the version information.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to display the forwarding table for LID 15 with the `ibroute` command.

Note – The output for your InfiniBand fabric will differ from that in the example.

```
# ibroute 15
Unicast lids [0x0-0xf] of switch Lid 15 guid 0x0021283a8389a0a0 (Sun DCS 36 QDR
switch localhost):
  Lid  Out   Destination
     Port     Info
0x000e 023 : (Channel Adapter portguid 0x0003ba000100e38a: 'mnm33-43 HCA-1')
```

```
0x000f 000 : (Switch portguid 0x0021283a8389a0a0: 'Sun DCS 36 QDR switch
localhost')
2 valid lids dumped
#
```

Related Information

- `ibroute` man page
- "ibtracert Command" on page 139
- "Linux Shells for InfiniBand Commands" on page 85

`ibrouters` Command

Displays InfiniBand router nodes in topology.

Syntax

`ibrouters [-h][`*topology*`|-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]`

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand software command is a script that discovers the InfiniBand fabric topology or uses an existing topology file to extract the router nodes.

The `ibrouters` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `ibrouters` command and their purposes:

Option	Purpose
`-h`	Provides help.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to display router nodes with the `ibrouters` command.

Note – The output in the example is a portion of the full output and varies for each InfiniBand topology.

```
# ibrouters
Router : 0x00066a00d80003fd ports 24 "Hellios 4024 Router " enhanced port 0 lid
7 lmc 0
#
```

Related Information

- `ibrouters` man page
- "ibnetdiscover Command" on page 118
- "ibnodes Command" on page 123
- "Linux Shells for InfiniBand Commands" on page 85

`ibstat` Command

Queries basic status of InfiniBand devices.

Syntax

`ibstat [-d][-e][-h][-l][-s][-p][-V]` *ca_name* `[`*ca_port*`]`

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.

Description

This InfiniBand software command displays basic information retrieved from the local InfiniBand driver. Output of the command includes:

- `LID`
- `SMLID`
- `port logical state`
- `link width`
- `port physical state`

The `ibstat` command is similar to the `ibstatus` command, however, the `ibstat` command is a binary executable, has options to display channel adapters and ports, and provides more information than the `ibstatus` command.

The `ibstat` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `ibstat` command and their purposes:

Option	Purpose
`-d`	Sets the debug level. Can be used several times to increase the debug level.
`-e`	Displays send and receive errors.
`-h`	Provides help.
`-l`	Lists all InfiniBand devices.
`-s`	Provides short output.
`-p`	Shows port list.

Option	Purpose
-V	Displays the version information.

Example

The following example shows how to display the basic status from the local InfiniBand driver with the `ibstat` command.

```
# ibstat
Switch 'is4_0'
  Switch type: MT48436
  Number of ports: 0
  Firmware version: 7.3.0
  Hardware version: a0
  Node GUID: 0x00212856cfe2c0a0
  System image GUID: 0x00212856cfe2c0a3
  Port 0:
      State: Active
      Physical state: LinkUp
      Rate: 40
      Base lid: 16
      LMC: 0
      SM lid: 16
      Capability mask: 0x4250084a
      Port GUID: 0x00212856cfe2c0a0
#
```

Related Information

- `ibstat` man page
- "ibstatus Command" on page 133
- "Linux Shells for InfiniBand Commands" on page 85

`ibstatus` Command

Queries basic status of InfiniBand devices.

Syntax

`ibstatus [-h][`*devname*`[:`*ib_port*`]]...`

where:

- *devname* is the InfiniBand device name.
- *ib_port* is the port number of the InfiniBand device.

Description

This InfiniBand command displays basic information retrieved from the local InfiniBand driver. Output of the command includes:

- `LID`
- `SMLID`
- `port logical state`
- `link width`
- `port physical state`

Note – This command is an alternative to the `ibstat` command.

Options

The `-h` option provides help.

Example

The following example shows how to display the basic status from the local InfiniBand driver with the `ibstatus` command.

```
# ibstatus
Infiniband device 'is4_0' port 0 status:
  default gid:     fe80:0000:0000:0000:0021:283a:8389:a0a0
  base lid:        0xf
  sm lid:          0xf
  state:           4: ACTIVE
```

```
  phys state:      5: LinkUp
  rate:            40 Gb/sec (4X QDR)
#
```

Related Information

- `ibstatus` man page

`ibswitches` Command

Displays InfiniBand switch node in the topology.

Syntax

`ibswitches [-h][`*topology*`|-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]`

where:

- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.

Description

This InfiniBand command is a script the discovers the InfiniBand fabric topology or uses an existing topology file to extract the switch nodes.

The `ibswitches` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `ibswitches` command and their purposes:

Option	Purpose
`-h`	Provides help.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to display the switch GUIDs with the `ibswitches` command.

Note – The output for your InfiniBand fabric will greatly differ from that in the example.

```
# ibswitches
Switch  : 0x00212856cfe2c0a0 ports 36 "SUN IB QDR GW switch mnm34-98.example.com"
enhanced port 0 lid 6 lmc 0
#
```

Related Information

- `ibswitches` man page
- "ibnetdiscover Command" on page 118
- "ibnodes Command" on page 123
- "Linux Shells for InfiniBand Commands" on page 85

`ibsysstat` Command

Displays system status of an InfiniBand address.

Syntax

`ibsysstat [-d][-e][-G][-h][-s` *smlid*`][-v][-V][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`][-o` *oui*`][-S]` *lid*|*guid* `[`*op*`]`

where:

- *smlid* is the Subnet Manager LID.
- *topology* is the topology file.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *oui* is the OUI number.
- *lid* is the local identifier.
- *guid* is the global unique identifier.
- *op* is the operation to perform on the node:
  - `ping` – Verify connectivity to the server.
  - `host` – Obtain host information from the server.
  - `cpu` – Obtain CPU information from the server.

Description

This InfiniBand command uses vendor supplied management datagrams to validate connectivity between InfiniBand nodes and return other information about the node. The command is a client-server, in that a remote node is configured as a server, while a local node performs as a client.

The command is first run locally on the server. The command is then run again locally on the client, pointing to the LID of the server. The ping repeats every second on the server, until you type the Ctrl-C key combination to exit the command.

Options

The following table describes the options to the `ibsysstat` command and their purposes:

Option	Purpose
`-d`	Sets the debug level. Can be used several times to increase the debug level.

Option	Purpose
-e	Displays send and receive errors.
-G	Uses the port GUID address.
-h	Provides help.
-s	Uses *smlid* as the target LID for Subnet Manager/Subnet Administrator queries.
-v	Provides verbose output.
-V	Displays the version information.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.
-o	Uses specified OUI number to multiplex vendor management datagrams.
-S	Starts in server mode.

Example

The following example shows how to retrieve system information about a remote node server from a local node client with the `ibsysstat` command. First configure the remote node server:

```
# ibsysstat -S
<CTRL-C to exit server>
#
```

Then run the command on the local node client:

```
# ibsysstat 15 ping
sysstat ping succeeded
# ibsysstat 15 host
mnm105-100.mnm.sfbay.sun.com
# ibsysstat 15 cpu
cpu 0: model  AMD Opteron(tm) Processor 848 MHZ  2189.680
cpu 1: model  AMD Opteron(tm) Processor 848 MHZ  2189.680
cpu 2: model  AMD Opteron(tm) Processor 848 MHZ  2189.680
cpu 3: model  AMD Opteron(tm) Processor 848 MHZ  2189.680
#
```

Related Information

- `ibsysstat` man page

`ibtracert` Command

Traces the InfiniBand path.

Syntax

`ibtracert [-d][-D][-G][-h][-m` *mlid*`][-s` *smlid*`][-v][-V][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`][`*lid*`|`*dr_path*`|`*guid* `[`*startlid* `[`*endlid*`]]]`

where:

- *mlid* is the multicast LID.
- *smlid* is the Subnet Manager LID.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *dr_path* is the directed path.
- *guid* is the global unique identifier.
- *startlid* is the starting LID for a range.
- *endlid* is the ending LID for a range.

Description

This InfiniBand command uses SMPs to trace the path from a source GID or LID to a destination GID or LID. Each responding hop in the path is displayed. The `-m` option enables multicast path tracing between source and destination nodes.

The `ibtracert` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `ibtracert` command and their purposes:

Option	Purpose
`-d`	Sets the debug level. Can be used several times to increase the debug level.
`-D`	Uses the directed path address. The path is a comma-delimited sequence of out ports.
`-G`	Uses the port GUID address.
`-h`	Provides help.
`-m`	Shows the multicast trace of the specified MLID.
`-s`	Uses *smlid* as the target LID for Subnet Manager or Subnet Administrator queries.
`-v`	Provides verbose output.
`-V`	Displays the version information.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to display the path from LID 25 to LID 24 with the `ibtracert` command.

Note – The output for your InfiniBand fabric will differ from that in the example.

```
# ibtracert 15 14
From switch {0x00212856cd22c0a0} portnum 0 lid 15-15 "SUN IB QDR GW switch
mnm34-97.example.com"
[1] -> ca port {0x00212856cd22c042}[2] lid 14-14 "SUN IB QDR GW switch
mnm34-97.example.com Bridge 1"
To ca {0x00212856cd22c040} portnum 2 lid 14-14 "SUN IB QDR GW switch
mnm34-97.example.com Bridge 1"
#
```

Related Information

- `ibtracert` man page
- "ibroute Command" on page 128
- "Linux Shells for InfiniBand Commands" on page 85

`perfquery` Command

Queries InfiniBand port counters.

Syntax

`perfquery [-d][-e][-G][-h][-a][-l][-r][-R][-v][-V][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`][`*lid*`|`*guid*` [[`*port*`][`*reset_mask*`]]`

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *guid* is the global unique identifier.
- *port* is the port being queried.
- *reset_mask* is the two-byte mask.

Description

This InfiniBand command uses the performance management GMPs to acquire the `PortCounters` or `PortExtendedCounters` from the Performance Manager agent at the node or port specified.

Note – The data values retrieved from `PortCounters` and `PortExtendedCounters` are represented as octets divided by 4.

Note – Providing a *port* value of 255 ensures that the operation is performed on all ports.

The `perfquery` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `perfquery` command and their purposes:

Option	Purpose
`-d`	Sets the debug level. Can be used several times to increase the debug level.
`-e`	Shows extended port counters.
`-a`	Shows aggregate counters for all ports.
`-G`	Uses the port GUID address.
`-h`	Provides help.
`-l`	Loops through all ports.
`-r`	Resets the counters after reading.
`-R`	Resets counters only.
`-v`	Provides verbose output.
`-V`	Displays the version information.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to display all port counters for LID 15, port 23 with the `perfquery` command.

```
# perfquery 15 23
# Port counters: Lid 15 port 23
PortSelect:......................23
```

```
CounterSelect:...................0x1b01
SymbolErrors:....................0
LinkRecovers:....................0
LinkDowned:......................0
RcvErrors:.......................0
RcvRemotePhysErrors:.............0
RcvSwRelayErrors:................0
XmtDiscards:.....................0
XmtConstraintErrors:.............0
RcvConstraintErrors:.............0
LinkIntegrityErrors:.............0
ExcBufOverrunErrors:.............0
VL15Dropped:.....................0
XmtData:.........................20232
RcvData:.........................20232
XmtPkts:.........................281
RcvPkts:.........................281
#
```

Related Information

- `perfquery` man page
- "ibcheckerrors Command" on page 88
- "ibdatacounters Command" on page 105
- "ibdatacounts Command" on page 106
- "Linux Shells for InfiniBand Commands" on page 85

`saquery` Command

Queries InfiniBand fabric administration attributes.

Syntax

`saquery [-h][-d][-p][-N][-D][-S][-I][-L][-l][-G][-O][-U][-c][-s][-g][-m][-x][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`][--src-to-dst` *source*`:`*destination*`][--sgid-to-dgid` *source-destination*`][`*name*`|`*lid*`|`*guid*`]`

where:

- *ca_name* is the channel adapter name.

- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *guid* is the global unique identifier.
- *name* is the query name described in the following table:

Query Names	Alias for *name*	Format
`ClassPortInfo`	`CPI`	
`NodeRecord`	`NR`	
`PortInfoRecord`	`PIR`	
`SL2VLTableRecord`	`SL2VL`	[[*lid*]/[*in_port*]/[*out_port*]]
`PKeyTableRecord`	`PKTR`	[[*lid*]/[*port*]/[*block*]]
`VLArbitrationTableRecord`	`VLAR`	[[*lid*]/[*port*]/[*block*]]
`InformInfoRecord`	`IIR`	
`LinkRecord`	`LR`	[[*from_lid*]/[*from_port*]] [[*to_lid*]/[*to_port*]]
`ServiceRecord`	`SR`	
`PathRecord`	`PR`	
`MCMemberRecord`	`MCMR`	
`LFTRecord`	`LFTR`	[[*lid*]/[*block*]]
`MFTRecord`	`MFTR`	[[*mlid*]/[*position*]/[*block*]]

Description

This InfiniBand command performs the selected Subnet Administrator query. Node records are queried by default.

The `saquery` command is available from the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `saquery` command and their purposes:

Option	Purpose
`-h`	Provides help.
`-d`	Sets the debug level. Can be used several times to increase the debug level.
`-p`	Displays the PathRecord information.
`-N`	Displays the NodeRecord information.
`-D`	Displays the NodeDescriptions of channel adapters only.
`-S`	Displays ServiceRecord information.
`-I`	Displays InformInfoRecord information.
`-L`	Returns the LIDs of the specified name.
`-l`	Returns the unique LID of the specified name.
`-G`	Returns the GUIDs of the specified name.
`-O`	Returns the name of the specified LID.
`-U`	Returns the name of the specified GUID.
`-G`	Uses the port GUID address.
`-c`	Displays the Subnet Administrator class port information.
`-s`	Returns the PortInforRecords with the `isSM` or `isSMdisabled` capability mask bit enabled.
`-g`	Displays multicast group information.
`-m`	Displays multicast member information. If a group is specified, provides only the GUID and node description for each entry.
`-x`	Displays LinkRecord information.
`--src-to-dst`	Displays a PathRecord for *source*:*destination*, where *source* and *destination* are either node names or LIDs.
`--sgid-to-dgid`	Displays a PathRecord for *source-destination*, where *source* and *destination* are GIDs in an IPv6 format acceptable to inet_pton.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to produce a node record dump of every LID in the InfiniBand fabric with the `saquery` command.

Note – The output in the example is a portion of the full output.

```
# saquery
NodeRecord dump:
     lid.....................0xE
     reserved................0x0
     base_version............0x1
     class_version...........0x1
     node_type...............Channel Adapter
     num_ports...............0x2
     sys_guid................0x0003ba000100e38b
     node_guid...............0x0003ba000100e388
     port_guid...............0x0003ba000100e38a
     partition_cap...........0x80
     device_id...............0x673C
     revision................0xA0
     port_num................0x2
     vendor_id...............0x2C9
     NodeDescription.........mnm33-43 HCA-1
NodeRecord dump:
     lid.....................0xF
     reserved................0x0
.
.
.
#
```

Related Information

- `saquery` man page
- "Linux Shells for InfiniBand Commands" on page 85

`sminfo` Command

Queries the InfiniBand `SMInfo` attribute.

Syntax

`sminfo [-d][-e] -s` *state* `-p` *priority* `-a` *activity* `[-D][-G][-h][-V][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]` *smlid*|*smdr_path*

where:

- *state* is the state for the Subnet Manager.
- *priority* is the priority.
- *activity* is the activity count.
- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *smlid* is the Subnet Manager local identifier.
- *smdr_path* is the directed path for the Subnet Manager.

Description

This InfiniBand command conducts a query of the Subnet Manager and outputs the information in a human readable format. The target Subnet Manager is identified in the local port information, or it is specified by the *smlid* or *smdr_path*.

Note – Using the `sminfo` command for other than simple queries might fault the target Subnet Manager.

Options

The following table describes the options to the `sminfo` command and their purposes:

Option	Purpose
`-d`	Sets the debug level. Can be used several times to increase the debug level.
`-D`	Uses the directed path address. The path is a comma-delimited sequence of out ports.
`-e`	Displays send and receive errors.

Option	Purpose
-s	Sets the Subnet Manager state: • 0 – Not active. • 1 – Discovering. • 2 – Standby. • 3 – Master.
-p	Sets the priority, (0–15).
-a	Sets the activity count.
-G	Uses the port GUID address.
-h	Provides help.
-V	Displays the version information.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to display the SMInfo with the `sminfo` command.

```
# sminfo
sminfo: sm lid 15 sm guid 0x21283a8389a0a0, activity count 32046 priority 8 state
3 SMINFO_MASTER
#
```

Related Information

- `sminfo` man page
- "getmaster Command" on page 39
- "smpdump Command" on page 148

smpdump Command

Dumps the InfiniBand fabric management attributes.

Syntax

`smpdump [-s][-D][-h][-V][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]`
lid|*dr_path* *attr* `[`*mod*`]`

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *lid* is the local identifier.
- *dr_path* is the directed path.
- *attr* is the InfiniBand architecture attribute ID for Subnet Manager attribute.
- *mod* is the InfiniBand architecture modifier for Subnet Manager attribute.

Description

This InfiniBand command is a general purpose SMP utility that returns Subnet Manager attributes from a specified SMA. The output is in hexadecimal.

Options

The following table describes the options to the `smpdump` command and their purposes:

Option	Purpose
`-D`	Uses the directed path address. The path is a comma delimited sequence of out ports.
`-e`	Displays send and receive errors.
`-g`	Shows the GID address only.
`-h`	Provides help.
`-V`	Displays the version information.
`-C`	Uses the specified channel adapter name.
`-P`	Uses the specified channel adapter port.
`-t`	Overrides the default timeout.

Example

The following example shows how to display a raw dump of the InfiniBand management attributes for the directed path through LIDs 14 and 15 with the `smpdump` command.

```
# smpdump 14 15
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000
SMP status: 0xc
#
```

Related Information

- `smpdump` man page
- "smpquery Command" on page 150

`smpquery` Command

Queries InfiniBand fabric management attributes.

Syntax

`smpquery [-d][-D][-e][-G][-h][-v][-V][-C` *ca_name*`][-P` *ca_port*`][-t` *timeout*`]` *op* *lid*|*dr_path*|*guid* `[`*params*`]`

where:

- *ca_name* is the channel adapter name.
- *ca_port* is the channel adapter port.
- *timeout* is the timeout in milliseconds.
- *op* is the supported operation.
- *lid* is the local identifier.
- *dr_path* is the directed path.
- *guid* is the global unique identifier.
- *params* is the parameter of the operation.

Description

This InfiniBand command permits a subset of the standard SMP queries, including the following:

- Node information
- Node description
- Switch information
- Port information

Output is in human-readable format.

Supported operations and parameters are as follows:

- `nodeinfo` *addr*
- `nodedesc` *addr*
- `portinfo` *addr* [*portnum*]
- `switchinfo` *addr*
- `pkeys` *addr* [*portnum*]
- `sl2vl` *addr* [*portnum*]
- `vlarb` *addr* [*portnum*]
- `guids` *addr*

where:

- *addr* is the address.
- *portnum* is the port number.

The `smpquery` command is available from the `/SYS/Switch_Diag`, `/SYS/Gateway_Mgmt`, and `/SYS/Fabric_Mgmt` Linux shell targets of the Oracle ILOM CLI interface.

Options

The following table describes the options to the `smpquery` command and their purposes:

Option	Purpose
`-d`	Sets the debug level. Can be used several times to increase the debug level.
`-D`	Uses the directed path address. The path is a comma-delimited sequence of out ports.
`-e`	Displays send and receive errors.

Option	Purpose
-G	Uses the port GUID address.
-h	Provides help.
-v	Provides verbose output.
-V	Displays the version information.
-C	Uses the specified channel adapter name.
-P	Uses the specified channel adapter port.
-t	Overrides the default timeout.

Example

The following example shows how to display node information for LID 15 with the `smpquery` command.

```
# smpquery nodeinfo 15
# Node info: Lid 15
BaseVers:.....................1
ClassVers:....................1
NodeType:.....................Switch
NumPorts:.....................36
SystemGuid:...................0x0021283a8389a0a3
Guid:.........................0x0021283a8389a0a0
PortGuid:.....................0x0021283a8389a0a0
PartCap:......................8
DevId:........................0xbd36
Revision:.....................0x000000a0
LocalPort:....................1
VendorId:.....................0x0002c9
#
```

Related Information

- `smpquery` man page
- "saquery Command" on page 143
- "smpdump Command" on page 148
- "Linux Shells for InfiniBand Commands" on page 85

Understanding SNMP MIB OIDs

These topics provide tables of object identifiers (OIDs) for their respective management information bases (MIBs).

Related Information

- "Understanding Hardware Commands" on page 1
- "Understanding InfiniBand Commands" on page 83

OID Tables Overview

Each OID table has three columns of information. From left to right, the columns are:

- **Description of Task or Activity** – The text under this heading describes what you want to do or information you want to see.
- **Action** – The verbs under this heading describe the action respective to the SNMP client software interface. For example, a verb of `read` means to use the `snmpget`, `snmpwalk`, or `snmptable` command in the NetSNMP CLI client.
- **MIB OID** – The string under this heading is the object identifier of the respective MIB that is used to accomplish the task or activity.

Note – The OID provided in the right column might not be complete. You might need to appended the OID with a `.0` if the OID is a scalar OID or a `.`*x* (where *x* is `1` to a maximum value) if the OID is a tabular OID.

A table OID has a `Table` suffix, and the OIDs listed beneath it are most likely to be columns of that table.

An index OID has an `Index` suffix, and while the OID is not directly accessible, the OID is still used to return a specific row of a table OID.

Related Information

- "Understanding the SUN-DCS-MIB MIB OIDs" on page 154
- "SUN-HW-TRAP-MIB MIB OIDs" on page 170
- "Understanding the SUN-ILOM-CONTROL-MIB MIB OIDs" on page 171
- "Understanding the SUN-PLATFORM-MIB MIB OIDs" on page 181
- "Understanding the ENTITY-MIB MIB OIDs" on page 192

Understanding the `SUN-DCS-MIB` MIB OIDs

This topic provides tables of the hardware, Subnet Manager, Performance Manager, and SNMP trap object identifiers.

Tables of object identifiers are provided in the following topics:

- "SUN-DCS-MIB MIB OID Command Syntax" on page 155
- "Enclosure OIDs" on page 156
- "Enclosure Nodes OIDs" on page 156
- "Neighbor Node OIDs" on page 157
- "Subnet Manager Info OIDs" on page 158
- "Subnet Manager Agent Data Port OIDs" on page 158
- "Understanding Performance Manager Agent OIDs" on page 160
- "Understanding SNMP Trap OIDs" on page 163
- "Understanding Gateway Specific OIDs" on page 167

Related Information

- "OID Tables Overview" on page 153
- "SUN-HW-TRAP-MIB MIB OIDs" on page 170
- "Understanding the SUN-ILOM-CONTROL-MIB MIB OIDs" on page 171

- "Understanding the SUN-PLATFORM-MIB MIB OIDs" on page 181
- "Understanding the ENTITY-MIB MIB OIDs" on page 192

SUN-DCS-MIB MIB OID Command Syntax

When you use the object identifiers in a command syntax, you must include the SUN-DCS-MIB MIB name. For example, using the V2c protocol:

```
$ command -v2c -c public mc_IP SUN-DCS-MIB::object_id argument
```

Or using the V3 protocol:

```
$ command -v3 -u snmp_user -l security_level -a authentication_protocol -A authentication_password
-x DES -X privacy_password mc_IP SUN-DCS-MIB::object_id argument
```

where:

- *command* is the command of the SNMP client that performs the action.
- *snmp_user* is a configured user of the SNMP services.
- *security_level* is:
  - noAuthNoPriv – There is no authentication or privacy.
  - authNoPriv – There is authentication, but no privacy.
  - authPriv – There is authentication and privacy.
- *authentication_protocol* is either MD5 or SHA
- *authentication_password* is the *snmp_user*'s authentication password.
- *privacy_password* is the *snmp_user*'s privacy password.
- *mc_IP* is the IP address of the management controller.
- *object_id* is the object identifier, as listed in the right column of the tables.
- *argument* is a combination of options and variables that support the object identifier.

For example, to securely display the version of the MIB on the management controller with IP address of 123.45.67.89:

```
$ snmpget -v3 -u usersnmp -l authPriv -a MD5 -A authpass -x DES -X privpass
123.45.67.89 SUN-DCS-MIB::mibVersion
```

Related Information

- "SUN-ILOM-CONTROL-MIB MIB OID Command Syntax" on page 172

- "SUN-PLATFORM-MIB MIB OID Command Syntax" on page 182
- "ENTITY-MIB MIB OID Command Syntax" on page 192

Enclosure OIDs

The following table lists enclosure object identifiers supported by the SUN-DCS-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display the MIB version.	read	mibVersion
Display the type of platform (Sun DCS GW, and so on).	read	platformName
Display the service processor firmware version.	read	spFwVersion
Display the FPGA/CPLD firmware version.	read	fpgaVersion

Related Information

- "OID Tables Overview" on page 153
- "SUN-DCS-MIB MIB OID Command Syntax" on page 155

Enclosure Nodes OIDs

The following table lists enclosure node object identifiers supported by the SUN-DCS-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display a table of information about the InfiniBand nodes belonging to the enclosure.	read	ibEncNodesTable
Display a row of the ibEncNodesTable.	read	ibEncNodesEntry
Display an index of InfiniBand nodes belonging to the enclosure.	read	ibEncNodesIndex
Display the type of InfiniBand node (switch, CA, and so on).	read	ibEncNodesType
Display the GUID of the node.	read	ibEncNodesGuid
Display the number of ports of the node.	read	ibEncNodesPorts
Display the NodeDescription of the node.	read	ibEncNodesDescr

Description of Task or Activity	Action	MIB OID
Display the LID of the node.	read	ibEncNodesLid
Display the firmware version of the node.	read	ibEncNodesFwVersion
Display the board or FRU on which the InfiniBand node is located.	read	ibEncNodesBoard
Display additional identification information of the InfiniBand node.	read	ibEncNodesName

Related Information

- "OID Tables Overview" on page 153
- "SUN-DCS-MIB MIB OID Command Syntax" on page 155

Neighbor Node OIDs

The following table lists neighbor node object identifiers supported by the SUN-DCS-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display a table of information about neighboring InfiniBand nodes.	read	ibNeighborTable
Display a row of the ibNodeNeighborTable.	read	ibNeighborEntry
Display an index of neighboring InfiniBand nodes.	read	ibNeighNodeIndex
Display an index of neighboring InfiniBand node data ports.	read	ibNeighPortIndex
Display the type of remote InfiniBand node (switch, CA, and so on).	read	ibNeighRemoteNodeType
Display the GUID of the remote node.	read	ibNeighRemoteNodeGuid
Display the number of ports of the remote node.	read	ibNeighRemoteNodePort
Display the NodeDescription of the remote node.	read	ibNeighRemoteNodeDescr
Display the LID of the remote node.	read	ibNeighRemoteNodeLid
Display a mapping of neighboring InfiniBand device port to connector.	read	ibNeighPortConnectorPosition

Related Information

- "OID Tables Overview" on page 153
- "SUN-DCS-MIB MIB OID Command Syntax" on page 155

Subnet Manager Info OIDs

The following table lists Subnet Manager object identifiers supported by the SUN-DCS-MIB MIB.

Description of Task or Activity	Action	MIB OID
Identify the master Subnet Manager in the subnet.	read	smMasterSMPresent
Display the LID of the master Subnet Manager.	read	smMasterSMLid
Display the GUID of the master Subnet Manager.	read	smMasterSMGuid
Display information about the Subnet Manager. Disable or enable the Subnet Manager.	read write	smLocalSMRunning
Display the state of the Subnet Manager.	read	smLocalSMState

Related Information

- "OID Tables Overview" on page 153
- "SUN-DCS-MIB MIB OID Command Syntax" on page 155

Subnet Manager Agent Data Port OIDs

The following table lists SMA object identifiers supported by the SUN-DCS-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display a table of SMA PortInfo information.	read	ibSmaPortInfoTable
Display a row of the ibSmaPortInfoTable.	read	ibSmaPortInfoEntry
Display an index of SMA InfiniBand nodes.	read	ibSmaNodeIndex
Display an index of SMA InfiniBand data ports.	read	ibSmaPortIndex
Display SMA enabled link width.	read	ibSmaPortLinkWidthEnabled
Display SMA supported link width.	read	ibSmaPortLinkWidthSupported

Description of Task or Activity	Action	MIB OID
Display SMA currently active link width.	read	ibSmaPortLinkWidthActive
Display SMA supported link speed.	read	ibSmaPortLinkSpeedSupported
Display the SMA link state of the port.	read	ibSmaPortLinkState
Display the SMA physical state of the port.	read	ibSmaPortPhysState
Display the SMA LinkDownDefault state.	read	ibSmaPortLinkDownDefaultState
Display the SMA LID mask control count for multipath support of CA and router ports.	read	ibSmaPortLidMaskCount
Display the currently active SMA link speed.	read	ibSmaPortLinkSpeedActive
Display the enabled SMA link speed.	read	ibSmaPortLinkSpeedEnabled
Display the active maximum MTU enabled on the SMA port for transmit.	read	ibSmaPortNeighborMtu
Display the virtual lane configuration supported on the SMA port.	read	ibSmaPortVirtLaneSupport
Display the number of high priority packets that are transmitted before a low priority packet is sent.	read	ibSmaPortVlHighPriorityLimit
Display the number of virtual lane - weight pairs supported on the SMA port in the ibSmaHiPriVlArbTable for high priority.	read	ibSmaPortVlArbHighCapacity
Display the number of virtual lane - weight pairs supported on the SMA port in the ibSmaHiPriVlArbTable for low priority.	read	ibSmaPortVlArbLowCapacity
Display the maximum MTU supported by the SMA port.	read	ibSmaPortMtuCapacity
Display or specify the number of sequential packets dropped that causes the SMA port to enter the VLStalled state.	read	ibSmaPortVlStallCount
Display or specify the time a packet can live at the head of a virtual lane queue.	read	ibSmaPortHeadOfQueueLife
Display the virtual lanes operational on the SMA port.	read	ibSmaPortOperationalVls
Display support of optional inbound partition enforcement.	read	ibSmaPortPartEnforceInbound
Display support of optional outbound partition enforcement.	read	ibSmaPortPartEnforceOutbound
Display support of optional inbound raw packet enforcement.	read	ibSmaPortFilterRawPktInbound

Description of Task or Activity	Action	MIB OID
Display support of optional outbound raw packet enforcement.	`read`	`ibSmaPortFilterRawPktOutbound`
Display the local physical error threshold value. When the threshold is exceeded, a local link integrity error is determined.	`read`	`ibSmaPortLocalPhysErrorThreshold`
Display overrun error threshold value. When the number of buffer overruns exceeds the threshold, an excessive buffer overrun error is determined.	`read`	`ibSmaPortOverrunErrorThreshold`
Display a mapping of an InfiniBand device SMA port to connector.	`read`	`ibSmaPortConnectorPosition`

Related Information

- "OID Tables Overview" on page 153
- "SUN-DCS-MIB MIB OID Command Syntax" on page 155

Understanding Performance Manager Agent OIDs

Performance Manager object identifiers are provided in these tables:

- "PMA Port Counters Table OIDs" on page 161
- "PMA Extended Port Counters Table OIDs" on page 162

Related Information

- "Understanding SNMP Trap OIDs" on page 163
- "Understanding Gateway Specific OIDs" on page 167
- "OID Tables Overview" on page 153
- "SUN-HW-TRAP-MIB MIB OIDs" on page 170
- "Understanding the SUN-ILOM-CONTROL-MIB MIB OIDs" on page 171
- "Understanding the SUN-PLATFORM-MIB MIB OIDs" on page 181
- "Understanding the ENTITY-MIB MIB OIDs" on page 192

PMA Port Counters Table OIDs

The following table lists PMA port counter object identifiers supported by the SUN-DCS-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display a table of PMA Port Counters information.	read	ibPmaPortCntrsTable
Display a row of the ibPmaPortCntrsTable	read	ibPmaPortCntrsEntry
Display an index of PMA nodes.	read	ibPmaPortCntrsNode
Display an index of the PMA InfiniBand data ports.	read	ibPmaPortCntrsPort
Display the total number of symbol errors detected on one or more physical lanes.	read	ibPmaSymbolErrCounter
Display the number of times the port training state machine has successfully completed the link error recovery process.	read	ibPmaLinkErrRecoveryCntr
Display the number of times the port training state machine has failed the link error recovery process and downed the link.	read	ibPmaLinkDownedCntr
Display the number of packets containing an error that were received on the port.	read	ibPmaPortRcvErr
Display the number of packets marked with the End-of-Bad-Packets (EBP) delimiter received on the port.	read	ibPmaPortRcvRemPhysErr
Display the number of packets received on the port that could not be forwarded by the switch relay.	read	ibPmaPortRcvSwitchRelayErr
Display the number of outbound packets discarded because the port is down or congested.	read	ibPmaPortXmitDiscard
Display the number of packets not transmitted from the port because the packet is raw, or fails partition key check or IP version check.	read	ibPmaPortXmitConstraintErr
Display the number of packets received on the port that are discarded because the packet is raw, or fails partition key check or IP version check.	read	ibPmaPortRcvConstraintErr
Display the number of times that the local physical errors threshold was exceeded.	read	ibPmaLocalLinkIntegrityErr

Description of Task or Activity	Action	MIB OID
Display the number of times that buffer overrun errors occurred.	read	ibPmaExcessBufOverrunErr
Display the number of incoming VL 15 packets dropped due to lack of buffers.	read	ibPmaVl15Dropped
Display a mapping of PMA InfiniBand device port to connector.	read	ibPmaPortConnector
Display the number of symbol errors for a port during the last 10 minutes.	read	ibPmaSymErrors10min
Display the number of symbol errors for a port during the last hour.	read	ibPmaSymErrors1hour
Display the number of symbol errors for a port during the last 24 hours.	read	ibPmaSymErrors24hours

Related Information

- "OID Tables Overview" on page 153
- "SUN-DCS-MIB MIB OID Command Syntax" on page 155

PMA Extended Port Counters Table OIDs

The following table lists extended PMA port counter object identifiers supported by the SUN-DCS-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display a table of PMA extended port counters information.	read	ibPmaExtPortCntrsTable
Display a row of the ibPmaExtPortCntrsTable.	read	ibPmaExtPortCntrsEntry
Display an index of external PMA nodes.	read	ibPmaExtPortCntrsNode
Display an index of external PMA InfiniBand data ports.	read	ibPmaExtPortCntrsPort
Display the number of data octets, divided by 4, transmitted on all virtual lanes from the port.	read	ibPmaPortXmitData
Display the number of data octets, divided by 4, received on all virtual lanes at the port.	read	ibPmaPortRcvData
Display the number of packets transmitted on all virtual lanes from the port.	read	ibPmaPortXmitPkts

Description of Task or Activity	Action	MIB OID
Display the number of packets received from all virtual lanes on the port.	read	ibPmaPortRcvPkts
Display the number of unicast packets transmitted on all virtual lanes from the port.	read	ibPmaPortUnicastXmitPkts
Display the number of unicast packets received from all virtual lanes on the port.	read	ibPmaPortUnicastRcvPkts
Display the number of multicast packets transmitted on all virtual lanes from the port.	read	ibPmaPortMulticastXmitPkts
Display the number of multicast packets received from all virtual lanes on the port.	read	ibPmaPortMulticastRcvPkts
Display a mapping of external PMA InfiniBand device port to connector.	read	ibPmaExtPortConnector

Related Information

- "OID Tables Overview" on page 153
- "SUN-DCS-MIB MIB OID Command Syntax" on page 155

Understanding SNMP Trap OIDs

SNMP trap object identifiers are provided in these tables:

- "Trap Control OIDs" on page 164
- "Trap Threshold OIDs" on page 164
- "Trap Notification OIDs" on page 166

Related Information

- "Understanding Performance Manager Agent OIDs" on page 160
- "Understanding Gateway Specific OIDs" on page 167
- "OID Tables Overview" on page 153
- "SUN-HW-TRAP-MIB MIB OIDs" on page 170
- "Understanding the SUN-ILOM-CONTROL-MIB MIB OIDs" on page 171
- "Understanding the SUN-PLATFORM-MIB MIB OIDs" on page 181
- "Understanding the ENTITY-MIB MIB OIDs" on page 192

Trap Control OIDs

The following table lists trap control object identifiers supported by the SUN-DCS-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display or set the state whether performance counters traps are generated.	read write	ibSnmpPerfCountersTrapEnable
Display or set whether master Subnet Manager traps are generated.	read write	ibSnmpMasterSMTrapEnable
Display or set whether link state and speed traps are generated.	read write	ibSnmpLinkTrapEnable
Display or set whether high error rate traps are generated.	read write	ibSnmpPerfHighErrorRateTrapEnable

Related Information

- "OID Tables Overview" on page 153
- "SUN-DCS-MIB MIB OID Command Syntax" on page 155

Trap Threshold OIDs

The following table lists trap threshold object identifiers supported by the SUN-DCS-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display or set whether a trap is generated, should the SymbolErrCounter threshold be exceeded.	read write	ibSnmpPerfSymbolErrCounterThreshold
Display or set whether a trap is generated, should the LinkErrRecoveryCntr threshold be exceeded.	read write	ibSnmpPerfLinkErrRecoveryCntrThreshold
Display or set whether a trap is generated, should the LinkDownedCntr threshold be exceeded.	read write	ibSnmpPerfLinkDownedCntrThreshold
Display or set whether a trap is generated, should the PortRcvErr threshold be exceeded.	read write	ibSnmpPerfPortRcvErrThreshold

Description of Task or Activity	Action	MIB OID
Display or set whether a trap is generated, should the PortRcvRemPhysErr threshold be exceeded.	read write	ibSnmpPerfPortRcvRemPhysErrThreshold
Display or set whether a trap is generated, should the PortRcvSwitchRelayErr threshold be exceeded.	read write	ibSnmpPerfPortRcvSwitchRelayErrThreshold
Display or set whether a trap is generated, should the PortXmitDiscards threshold be exceeded.	read write	ibSnmpPerfPortXmitDiscardThreshold
Display or set whether a trap is generated, should the PortXmitConstraintErr threshold be exceeded.	read write	ibSnmpPerfPortXmitConstraintErrThreshold
Display or set whether a trap is generated, should the PortRcvConstraintErr threshold be exceeded.	read write	ibSnmpPerfPortRcvConstraintErrThreshold
Display or set whether a trap is generated, should the LocalLinkIntegrityErr threshold be exceeded.	read write	ibSnmpPerfLocalLinkIntegrityErrThreshold
Display or set whether a trap is generated, should the ExcessBufOverrunErr threshold be exceeded.	read write	ibSnmpPerfExcessBufOverrunErrThreshold
Display or set whether a trap is generated, should the Vl15Dropped threshold be exceeded.	read write	ibSnmpPerfVl15DroppedThreshold

Related Information

- "OID Tables Overview" on page 153
- "SUN-DCS-MIB MIB OID Command Syntax" on page 155

Trap Notification OIDs

The following table lists trap notification object identifiers supported by the SUN-DCS-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display an index of InfiniBand nodes originating a trap.	read	ibSnmpTrapNodeIndex
Display an index of InfiniBand data ports originating a trap.	read	ibSnmpTrapPortIndex
Display the performance counter causing a trap.	read	ibSnmpTrapPerfErrCounter
Display the counter value causing a trap.	read	ibSnmpTrapCounterVal
Display a mapping of SNMP InfiniBand device port to connector causing a trap.	read	ibSnmpTrapPortConnector
Identify the master Subnet Manager causing a trap.	read	ibSnmpTrapMasterSMPresent
Display the LID of the master Subnet Manager of a trap.	read	ibSnmpTrapMasterSMLid
Display the currently active link width of a trap.	read	ibSnmpTrapLinkWidthActive
Display the state of the link on the port of a trap.	read	ibSnmpTrapLinkState
Display the currently active link speed of a trap.	read	ibSnmpTrapLinkSpeedActive
Display the GUID of the trap's node.	read	ibSnmpTrapNodeGuid
Display the LID of the trap's node.	read	ibSnmpTrapNodeLid
Display the NodeDescription of the trap's node.	read	ibSnmpTrapNodeDescr
Display how much the symbol error counter increased during the error rate monitoring interval.	read	ibSnmpSymErrIncrease
Display and set the error rate monitoring interval.	read write	ibSnmpErrRateInterval
A performance error counter threshold was exceeded. Node, port, counter and its value are passed inside the trap.		ibSnmpPerfErrCounterTrap

Description of Task or Activity	Action	MIB OID
This trap is sent when a change happens in master Subnet Manager presence.		`ibSnmpNoMasterSMTrap`
This trap is sent when a change in `LinkState`, `LinkSpeedActive`, or `LinkWidthActive` happens on a port.		`ibSnmpLinkTrap`
This trap is sent when error rate on a port is high (according to the configured values).		`ibSnmpHighErrRateTrap`

Related Information

- "OID Tables Overview" on page 153
- "SUN-DCS-MIB MIB OID Command Syntax" on page 155

Understanding Gateway Specific OIDs

Gateway specific object identifiers are provided in these tables:

- "Gateway Port Status OIDS" on page 168
- "Gateway Port Counter OIDS" on page 168

Related Information

- "Understanding Performance Manager Agent OIDs" on page 160
- "Understanding SNMP Trap OIDs" on page 163
- "OID Tables Overview" on page 153
- "SUN-HW-TRAP-MIB MIB OIDs" on page 170
- "Understanding the SUN-ILOM-CONTROL-MIB MIB OIDs" on page 171
- "Understanding the SUN-PLATFORM-MIB MIB OIDs" on page 181
- "Understanding the ENTITY-MIB MIB OIDs" on page 192

Gateway Port Status OIDS

The following table lists gateway port status object identifiers supported by the SUN-DCS-MIB MIB.

Description of Task or Activity	Action	MIB OID
Displays a table of gateway port information.	read	gwPortStateTable
Displays a row of the gwPortStateTable.	read	gwPortStateEntry
Display the gateway's port's short names.	read	gwPortShortName
Display the gateway's port's long names.	read	gwPortLongName
Display the BridgeX manager's port's state.	read	gwPortState
Display the BridgeX manager's port's physical link state.	read	gwPortLinkState
Identifies that a port is enabled.	read	gwPortEnabled
Display the port protocol as either Ethernet or Fiber Channel.	read	gwPortProtocol
Display the Ethernet protocol.	read	gwPortEthProto
Display the supported Ethernet protocols.	read	gwPortEthSupportProto
Display the Ethernet MTU.	read	gwPortEthMTU
Display the Ethernet transmit pause.	read	gwPortEthTxPause
Display the Ethernet receive pause.	read	gwPortEthRxPause

Related Information

- "OID Tables Overview" on page 153
- "SUN-DCS-MIB MIB OID Command Syntax" on page 155

Gateway Port Counter OIDS

The following table lists gateway port counter object identifiers supported by the SUN-DCS-MIB MIB.

Description of Task or Activity	Action	MIB OID
Displays a table of gateway port counters.	read	gwEthPortCntrsTable
Displays a row of the gwEthPortCntrsTable.	read	gwEthPortCntrsEntry
Display the gateway's port's short name.	read	gwEthPortName

Description of Task or Activity	Action	MIB OID
Display the Ethernet bytes received counter.	read	gwEthRxBytes
Display the Ethernet packets received counter.	read	gwEthRxPkts
Display the Ethernet Jumbo packets received counter.	read	gwEthRxJumboPkts
Display the Ethernet unicast packets received counter.	read	gwEthRxUcastPkts
Display the Ethernet multicast packets received counter.	read	gwEthRxMcastPkts
Display the Ethernet broadcast packets received counter.	read	gwEthRxBcastPkts
Display the Ethernet No Buffer received counter.	read	gwEthRxNoBuffer
Display the Ethernet CRC received counter.	read	gwEthRxCRC
Display the Ethernet runt received counter.	read	gwEthRxRunt
Display the Ethernet errors received counter.	read	gwEthRxErrors
Display the Ethernet bytes transmitted counter.	read	gwEthTxBytes
Display the Ethernet packets transmitted counter.	read	gwEthTxPkts
Display the Ethernet Jumbo packets transmitted counter.	read	gwEthTxJumboPkts
Display the Ethernet unicast packets transmitted counter.	read	gwEthTxUcastPkts
Display the Ethernet multicast packets transmitted counter.	read	gwEthTxMcastPkts
Display the Ethernet broadcast packets transmitted counter.	read	gwEthTxBcastPkts
Display the Ethernet errors transmitted counter.	read	gwEthTxErrors

Related Information

- "OID Tables Overview" on page 153
- "SUN-DCS-MIB MIB OID Command Syntax" on page 155

SUN-HW-TRAP-MIB MIB OIDs

The following table lists hardware trap configuration object identifiers supported by the SUN-HW-TRAP-MIB MIB

Description of Task or Activity	Action	MIB OID
Display or set the text string used to identify the source of the trap. Typically the host name.	read	sunHwTrapSystemIdentifier
Display the name of the component which is the source of the trap.	read	sunHwTrapComponentName
Display the threshold type that the sensor is reporting.	read	sunHwTrapThresholdType
Display the threshold value that has been exceeded.	read	sunHwTrapThresholdValue
Display the threshold sensor's reading at the time of the trap.	read	sunHwTrapSensorValue
Display or set an optional description which provides additional information.	read	sunHwTrapAdditionalInfo
Display the object in the entPhysicalTable to which the notification applies.	read	sunHwTrapAssocObjectId
Display an indication of the severity of the notification.	read	sunHwTrapSeverity
Display the text string containing the chassis serial number.	read	sunHwTrapChassisId
Display the text string containing the product name.	read	sunHwTrapProductName
Display an indication as to why the component was disabled.	read	sunHwTrapDisableReason
Display or set the message used by the test trap.	read	sunHwTrapTestMessage

Related Information

- "OID Tables Overview" on page 153
- "Understanding the SUN-DCS-MIB MIB OIDs" on page 154
- "Understanding the SUN-ILOM-CONTROL-MIB MIB OIDs" on page 171
- "Understanding the SUN-PLATFORM-MIB MIB OIDs" on page 181
- "Understanding the ENTITY-MIB MIB OIDs" on page 192

Understanding the SUN-ILOM-CONTROL-MIB MIB OIDs

This topic provides many tables of the Oracle ILOM object identifiers.

Tables of object identifiers are provided in the following topics:

Related Information

- "OID Tables Overview" on page 153
- "Understanding the SUN-DCS-MIB MIB OIDs" on page 154
- "SUN-HW-TRAP-MIB MIB OIDs" on page 170
- "Understanding the SUN-PLATFORM-MIB MIB OIDs" on page 181
- "Understanding the ENTITY-MIB MIB OIDs" on page 192

SUN-ILOM-CONTROL-MIB MIB OID Command Syntax

When you use the object identifiers in a command syntax, you must include the SUN-ILOM-CONTROL-MIB MIB name. For example, using the V2c protocol:

```
$ command -v2c -c public mc_IP SUN-ILOM-CONTROL-MIB::object_id argument
```

Or using the V3 protocol:

```
$ command -v3 -u snmp_user -l security_level -a authentication_protocol -A authentication_password
-x DES -X privacy_password mc_IP SUN-ILOM-CONTROL-MIB::object_id argument
```

where:

- *command* is the command of the SNMP client that performs the action.
- *snmp_user* is a configured user of the SNMP services.
- *security_level* is:
  - noAuthNoPriv – There is no authentication or privacy.
  - authNoPriv – There is authentication, but no privacy.
  - authPriv – There is authentication and privacy.
- *authentication_protocol* is either MD5 or SHA
- *authentication_password* is the *snmp_user*'s authentication password.
- *privacy_password* is the *snmp_user*'s privacy password.
- *mc_IP* is the IP address of the management controller.
- *object_id* is the object identifier, as listed in the right column of the tables.
- *argument* is a combination of options and variables that support the object identifier.

For example, to securely display a table of information about the current local users on the management controller with IP address of 123.45.67.89:

```
$ snmpwalk -v3 -u usersnmp -l authPriv -a MD5 -A authpass -x DES -X privpass
123.45.67.89 SUN-ILOM-CONTROL-MIB::ilomCtrlLocalUserTable
```

Related Information

- "SUN-DCS-MIB MIB OID Command Syntax" on page 155
- "SUN-PLATFORM-MIB MIB OID Command Syntax" on page 182
- "ENTITY-MIB MIB OID Command Syntax" on page 192

NTP OIDs

The following table lists network time protocol object identifiers supported by the SUN-ILOM-CONTROL-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display or set the IP address of the first NTP server used by the device.	read write	ilomCtrlDeviceNTPServerOneIP
Display or set the IP address of the second NTP server used by the device.	read write	ilomCtrlDeviceNTPServerTwoIP

Related Information

- "OID Tables Overview" on page 153
- "SUN-ILOM-CONTROL-MIB MIB OID Command Syntax" on page 172

Remote Syslog OIDs

The following table lists remote syslog object identifiers supported by the SUN-ILOM-CONTROL-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display or set the IP address of the first remote syslog destination (log host).	read write	ilomCtrlRemoteSyslogDest1
Display or set the IP address of the second remote syslog destination (log host).	read write	ilomCtrlRemoteSyslogDest2

Related Information

- "OID Tables Overview" on page 153
- "SUN-ILOM-CONTROL-MIB MIB OID Command Syntax" on page 172

HTTP OIDs

The following table lists HTTP service object identifiers supported by the SUN-ILOM-CONTROL-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display or set whether or not the embedded web server is running and listening on the HTTP port.	read write	ilomCtrlHttpEnabled
Display or set the port number that the embedded web server should listen to for HTTP requests.	read write	ilomCtrlHttpPortNumber
Display or set whether or not the embedded web server should redirect HTTP connections to HTTPS.	read write	ilomCtrlHttpSecureRedirect

Related Information

- "OID Tables Overview" on page 153
- "SUN-ILOM-CONTROL-MIB MIB OID Command Syntax" on page 172

HTTPS OIDs

The following table lists HTTPS service object identifiers supported by the SUN-ILOM-CONTROL-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display or set whether or not the embedded web server is running and listening on the HTTPS port.	read write	ilomCtrlHttpsEnabled
Display or set the port number that the embedded web server should listen to for HTTPS requests.	read write	ilomCtrlHttpsPortNumber

Related Information

- "OID Tables Overview" on page 153
- "SUN-ILOM-CONTROL-MIB MIB OID Command Syntax" on page 172

Network OIDs

The following table lists network object identifiers supported by the SUN-ILOM-CONTROL-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display a table of targets whose networks can be controlled.	read	ilomCtrlNetworkTable
Display information about a target which can be reset.	read	ilomCtrlNetworkEntry
Display the nomenclature name for a target which has a configurable network.	read	ilomCtrlNetworkTarget
Display the MAC address of the service processor or system controller.	read	ilomCtrlNetworkMacAddress
Display whether the current target is configured to have static IP settings or whether these settings are retrieved dynamically from DHCP.	read	ilomCtrlNetworkIpDiscovery
Display the current IP address for the given target.	read	ilomCtrlNetworkIpAddress
Display the current IP gateway for the given target.	read	ilomCtrlNetworkIpGateway
Display the current IP netmask for the given target.	read	ilomCtrlNetworkIpNetmask
Display or set the pending value for the mode of IP discovery for the given target.	read write	ilomCtrlNetworkPendingIpDiscovery
Display or set the pending IP address for the given target.	read write	ilomCtrlNetworkPendingIpAddress
Display or set the pending IP gateway for the given target.	read write	ilomCtrlNetworkPendingIpGateway
Display or set the pending IP netmask for the given target.	read write	ilomCtrlNetworkPendingIpNetmask
Commit pending properties.	read write	ilomCtrlNetworkCommitPending
Display or set the pending management port for the given target.	read write	ilomCtrlNetworkPendingManagementPort

Description of Task or Activity	Action	MIB OID
Display or set the current management port for the given target.	read write	ilomCtrlNetworkManagementPort
Display the address of the DHCP server for the given target.	read	ilomCtrlNetworkDHCPServerAddr
Display or set whether or not the parameters are enabled.	read write	ilomCtrlNetworkState

Related Information

- "OID Tables Overview" on page 153
- "SUN-ILOM-CONTROL-MIB MIB OID Command Syntax" on page 172

User OIDs

The following table lists user object identifiers supported by the SUN-ILOM-CONTROL-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display a table of information about the current local users with their password state.	read	ilomCtrlLocalUserTable
Display information about a local user in the database.	read	ilomCtrlLocalUserEntry
Display the user name of a local user on the device.	read	ilomCtrlLocalUserUsername
Set the password of a local user on the device.	read write	ilomCtrlLocalUserPassword
Display or set the role that is associated with a user.	read write	ilomCtrlLocalUserRoles
Create a new user or delete an existing user.	write	ilomCtrlLocalUserRowStatus
Display or set the CLI mode a user is configured.	read write	ilomCtrlLocalUserCLIMode

Related Information

- "OID Tables Overview" on page 153
- "SUN-ILOM-CONTROL-MIB MIB OID Command Syntax" on page 172

Session OIDs

The following table lists session object identifiers supported by the SUN-ILOM-CONTROL-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display a table of information about the current user sessions.	read	ilomCtrlSessionsTable
Display the user name of the user associated with the session.	read	ilomCtrlSessionsUsername
Display the type of connection that the given user is using to access the device.	read	ilomCtrlSessionsConnectionType
Display the date and time that the user logged into the device.	read	ilomCtrlSessionsLoginTime

Related Information

- "OID Tables Overview" on page 153
- "SUN-ILOM-CONTROL-MIB MIB OID Command Syntax" on page 172

Event Log OIDs

The following table lists event log object identifiers supported by the SUN-ILOM-CONTROL-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display a table of information about the current entries in the event log.	read	ilomCtrlEventLogTable
Display an integer representing the type of event.	read	ilomCtrlEventLogType
Display the date and time that the event log entry was recorded.	read	ilomCtrlEventLogTimestamp
Display an integer representing the class of event.	read	ilomCtrlEventLogClass
Display the event severity corresponding to the given log entry.	read	ilomCtrlEventLogSeverity
Display the description of the event.	read	ilomCtrlEventLogDescription
Clear the event log.	write	ilomCtrlEventLogClear

Related Information

- "OID Tables Overview" on page 153
- "SUN-ILOM-CONTROL-MIB MIB OID Command Syntax" on page 172

Alert OIDs

The following table lists alert object identifiers supported by the `SUN-ILOM-CONTROL-MIB` MIB.

Description of Task or Activity	Action	MIB OID
Display a table of information used to view and add alert rules.	`read`	`ilomCtrlAlertsTable`
Display or set the minimum event severity which should trigger an alert, for a given class.	`read` `write`	`ilomCtrlAlertSeverity`
Display or set the type of notification for a given alert.	`read` `write`	`ilomCtrlAlertType`
Display or set the IP address to receive alert notifications when the alert type is `snmptrap(2)`, `ipmipet(3)`, or `remotesyslog(4)`.	`read` `write`	`ilomCtrlAlertDestinationIP`
Display or set the email address to send alert notifications when the alert type is `email(1)`.	`read` `write`	`ilomCtrlAlertDestinationEmail`
Display or set the version of SNMP trap that is used for the given alert rule.	`read` `write`	`ilomCtrlAlertSNMPVersion`
Display or set the community string to be used when the `ilomCtrlAlertSNMPVersion` property is set to `v1` or `v2c`.	`read` `write`	`ilomCtrlAlertSNMPCommunityOrUsername`
Display or set the destination port for SNMP traps.	`read` `write`	`ilomCtrlAlertDestinationPort`
Display or set the class name to filter emailed alerts on.	`read` `write`	`ilomCtrlAlertEmailEventClassFilter`
Display or set the type name to filter emailed alerts on.	`read` `write`	`ilomCtrlAlertEmailEventTypeFilter`
Display or set an optional format to identify the sender or the from address.	`read` `write`	`ilomCtrlAlertEmailCustomSender`
Display or set an optional string added to the beginning of the message body.	`read` `write`	`ilomCtrlAlertEmailMessagePrefix`

Related Information

- "OID Tables Overview" on page 153
- "SUN-ILOM-CONTROL-MIB MIB OID Command Syntax" on page 172

Clock OIDs

The following table lists clock object identifiers supported by the SUN-ILOM-CONTROL-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display or set the date and time of the device.	read write	ilomCtrlDateAndTime
Display or set whether or not Network Time Protocol is enabled.	read write	ilomCtrlNTPEnabled
Display or set the configured time zone string.	read write	ilomCtrlTimezone

Related Information

- "OID Tables Overview" on page 153
- "SUN-ILOM-CONTROL-MIB MIB OID Command Syntax" on page 172

Backup and Restore OIDs

The following table lists backup and restore object identifiers supported by the SUN-ILOM-CONTROL-MIB MIB.

Description of Task or Activity	Action	MIB OID
Set the target destination of a configuration XML file during backup and restore. The syntax is: {protocol}://[user:passwword]@]host[/][path/][file].	write	ilomCtrlBackupAndRestoreTargetURI

Description of Task or Activity	Action	MIB OID
Set the passphrase for encrypting or decrypting sensitive data during backup and restore.	write	ilomCtrlBackupAndRestorePassphrase
Set the action to backup or restore.	read write	ilomCtrlBackupAndRestoreAction
Display the current status of backup or restore.	read	ilomCtrlBackupAndRestoreActionStatus

Related Information

- "OID Tables Overview" on page 153
- "SUN-ILOM-CONTROL-MIB MIB OID Command Syntax" on page 172

Identification OIDs

The following table lists identification object identifiers supported by the SUN-ILOM-CONTROL-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display or set the system identifier, which is often the host name of the server associated with Oracle ILOM to be sent out in the varbind for all traps that Oracle ILOM generates.	read write	ilomCtrlSystemIdentifier
Display or set the host name for Oracle ILOM.	read write	ilomCtrlHostName

Related Information

- "OID Tables Overview" on page 153
- "SUN-ILOM-CONTROL-MIB MIB OID Command Syntax" on page 172

SMTP OIDs

The following table lists SMTP object identifiers supported by the SUN-ILOM-CONTROL-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display or set whether or not the SMTP client is enabled.	read write	ilomCtrlSMTPEnabled
Display or set the IP address of the SMTP server used as a name service for user accounts.	read write	ilomCtrlSMTPServerIP
Display or set the port number for the SMTP client.	read write	ilomCtrlSMTPPortNumber
Display or set an optional format to identify the sender or the from address.	read write	ilomCtrlSMTPCustomSender

Related Information

- "OID Tables Overview" on page 153
- "SUN-ILOM-CONTROL-MIB MIB OID Command Syntax" on page 172

Understanding the SUN-PLATFORM-MIB MIB OIDs

This topic provides tables of the hardware, sensor, alarms, and watchdog object identifiers.

Tables of object identifiers are provided in the following topics:

- "SUN-PLATFORM-MIB MIB OID Command Syntax" on page 182
- "Understanding MIB Physical OIDs" on page 183
- "MIB Trap OIDs" on page 190

Related Information

- "OID Tables Overview" on page 153
- "Understanding the SUN-DCS-MIB MIB OIDs" on page 154
- "SUN-HW-TRAP-MIB MIB OIDs" on page 170

- "Understanding the SUN-ILOM-CONTROL-MIB MIB OIDs" on page 171
- "Understanding the ENTITY-MIB MIB OIDs" on page 192

SUN-PLATFORM-MIB MIB OID Command Syntax

When you use the object identifiers in a command syntax, you must include the SUN-PLATFORM-MIB MIB name. For example, using the V2c protocol:

```
$ command -v2c -c public mc_IP SUN-PLATFORM-MIB::object_id argument
```

Or using the V3 protocol:

```
$ command -v3 -u snmp_user -l security_level -a authentication_protocol -A authentication_password
-x DES -X privacy_password mc_IP SUN-PLATFORM-MIB::object_id argument
```

where:

- *command* is the command of the SNMP client that performs the action.
- *snmp_user* is a configured user of the SNMP services.
- *security_level* is:
  - noAuthNoPriv – There is no authentication or privacy.
  - authNoPriv – There is authentication, but no privacy.
  - authPriv – There is authentication and privacy.
- *authentication_protocol* is either MD5 or SHA
- *authentication_password* is the *snmp_user*'s authentication password.
- *privacy_password* is the *snmp_user*'s privacy password.
- *mc_IP* is the IP address of the management controller.
- *object_id* is the object identifier, as listed in the right column of the tables.
- *argument* is a combination of options and variables that support the object identifier.

For example, to securely display a table of switch components addressable by the management controller with IP address of 123.45.67.89:

```
$ snmpwalk -v3 -u usersnmp -l authPriv -a MD5 -A authpass -x DES -X privpass
123.45.67.89 SUN-PLATFORM-MIB::sunPlatEquipmentTable
```

Related Information

- "SUN-DCS-MIB MIB OID Command Syntax" on page 155

- "SUN-ILOM-CONTROL-MIB MIB OID Command Syntax" on page 172
- "ENTITY-MIB MIB OID Command Syntax" on page 192

Understanding MIB Physical OIDs

Sensor and hardware information object identifiers are provided in these tables:

- "High-Level OIDs" on page 183
- "NIM Equipment Table OIDs" on page 184
- "Physical Class Extension Table OIDs" on page 185
- "Sun Platform Sensor Table OIDs" on page 185
- "Sun Platform Binary Sensor Table OIDs" on page 186
- "Sun Platform Numeric Sensor Table OIDs" on page 187
- "Discrete Sensor Table OIDs" on page 188
- "Discrete Sensor States Table OIDs" on page 189
- "Sun Platform Fan Table OIDs" on page 189
- "Sun Platform Alarm Table OIDs" on page 190

Related Information

- "MIB Trap OIDs" on page 190
- "OID Tables Overview" on page 153
- "Understanding the SUN-DCS-MIB MIB OIDs" on page 154
- "SUN-HW-TRAP-MIB MIB OIDs" on page 170
- "Understanding the SUN-ILOM-CONTROL-MIB MIB OIDs" on page 171
- "Understanding the ENTITY-MIB MIB OIDs" on page 192

High-Level OIDs

The following table lists the high level object identifier supported by the `SUN-PLATFORM-MIB` MIB.

Description of Task or Activity	Action	MIB OID
Display the time at which the agent was last started.	`read`	`sunPlatStartTime`

Related Information

- "OID Tables Overview" on page 153
- "SUN-PLATFORM-MIB MIB OID Command Syntax" on page 182

NIM Equipment Table OIDs

The following table lists NIM equipment object identifiers supported by the `SUN-PLATFORM-MIB` MIB.

Description of Task or Activity	Action	MIB OID
Display the NIM equipment table.	`read`	`sunPlatEquipmentTable`
Display information about a particular piece of equipment within the network element of the type specified by `entPhysicalClass`.	`read`	`sunPlatEquipmentEntry`
Display or set the administrative state of the managed object.	`read` `write`	`sunPlatEquipmentAdministrativeState`
Display the operational state of the managed object.	`read`	`sunPlatEquipmentOperationalState`
Display the alarm status of the managed object.	`read`	`sunPlatEquipmentAlarmStatus`
Display the unknown or known status of the managed object.	`read`	`sunPlatEquipmentUnknownStatus`
Display or set the specific or general location name of the component.	`read` `write`	`sunPlatEquipmentLocationName`

Related Information

- "OID Tables Overview" on page 153
- "SUN-PLATFORM-MIB MIB OID Command Syntax" on page 182

Physical Class Extension Table OIDs

The following table lists physical class extension object identifiers supported by the SUN-PLATFORM-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display a table of information about the physical class of the entity if it is classified as other(1).	read	sunPlatPhysicalTable
Display information about a piece of equipment with an entPhysicalClass of other(1).	read	sunPlatPhysicalEntry
Display possible physical classes where entPhysicalClass is specified as other(1).	read	sunPlatPhysicalClass

Related Information

- "OID Tables Overview" on page 153
- "SUN-PLATFORM-MIB MIB OID Command Syntax" on page 182

Sun Platform Sensor Table OIDs

The following table lists sensor object identifiers supported by the SUN-PLATFORM-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display a table of attributes common to all sensors.	read	sunPlatSensorTable
Display information about a piece of equipment with an entPhysicalClass of sensor(8).	read	sunPlatSensorEntry
Display possible sensor classes where entPhysicalClass is specified as sensor(8).	read	sunPlatSensorClass
Display possible sensor types where entPhysicalClass is specified as sensor(8).	read	sunPlatSensorType
Display or set the update interval for the sensor measured in milliseconds.	read	sunPlatSensorLatency

Related Information

- "OID Tables Overview" on page 153
- "SUN-PLATFORM-MIB MIB OID Command Syntax" on page 182

Sun Platform Binary Sensor Table OIDs

The following table lists binary sensor object identifiers supported by the `SUN-PLATFORM-MIB` MIB.

Description of Task or Activity	Action	MIB OID
Display a table of attributes specific to binary sensors.	`read`	`sunPlatBinarySensorTable`
Display information about a sensor with a `sunPlatSensorClass` value of `binary(1)`.	`read`	`sunPlatBinarySensorEntry`
Display the current state of the sensor.	`read`	`sunPlatBinarySensorCurrent`
Display the expected state of the sensor.	`read`	`sunPlatBinarySensorExpected`
Display or set the interpretation to be applied to a sensor reading of true.	`read`	`sunPlatBinarySensorInterpretTrue`
Display or set the interpretation to be applied to a sensor reading of false.	`read`	`sunPlatBinarySensorInterpretFalse`

Related Information

- "OID Tables Overview" on page 153
- "SUN-PLATFORM-MIB MIB OID Command Syntax" on page 182

Sun Platform Numeric Sensor Table OIDs

The following table lists numeric sensor object identifiers supported by the SUN-PLATFORM-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display a table of attributes specific to numeric sensors.	read	sunPlatNumericSensorTable
Display information about a sensor with a sunPlatSensorClass value of numeric(2).	read	sunPlatNumericSensorEntry
Display the base unit of the value returned by the sensor. For example, volts.	read	sunPlatNumericSensorBaseUnits
Display the exponent of 10 to be multiplied by the base unit. For example, 2. So the multiplier is 10^2 or 100.	read	sunPlatNumericSensorExponent
Display or set the rate associated with the base units, if any. For example, per second.	read	sunPlatNumericSensorRateUnits
Display the current reading of the sensor.	read	sunPlatNumericSensorCurrent
Display or set the lowest expected value from the sensor.	read	sunPlatNumericSensorNormalMin
Display the highest expected value from the sensor.	read	sunPlatNumericSensorNormalMax
Display or set the accuracy of the sensor expressed as a +/-% value in units of 100ths of a percent.	read	sunPlatNumericSensorAccuracy
Display or set the lower threshold at which a noncritical condition occurs.	read write	sunPlatNumericSensorLowerThresholdNonCritical
Display or set the upper threshold at which a noncritical condition occurs.	read write	sunPlatNumericSensorUpperThresholdNonCritical
Display or set the lower threshold at which a critical condition occurs.	read write	sunPlatNumericSensorLowerThresholdCritical

Description of Task or Activity	Action	MIB OID
Display or set the upper threshold at which a critical condition occurs.	read write	sunPlatNumericSensorUpperThresholdCritical
Display or set the lower threshold at which a fatal condition occurs.	read write	sunPlatNumericSensorLowerThresholdFatal
Display or set the upper threshold at which a fatal condition occurs.	read write	sunPlatNumericSensorUpperThresholdFatal
Display the hysteresis around the thresholds.	read	sunPlatNumericSensorHysteresis
Display or enable the thresholds of the sensor.	read write	sunPlatNumericSensorEnabledThresholds
Set the sensors thresholds to the default values.	read write	sunPlatNumericSensorRestoreDefaultThresholds

Related Information

- "OID Tables Overview" on page 153
- "SUN-PLATFORM-MIB MIB OID Command Syntax" on page 182

Discrete Sensor Table OIDs

The following table lists discrete sensor object identifiers supported by the SUN-PLATFORM-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display the discrete sensor table.	read	sunPlatDiscreteSensorTable
Display information about a discrete sensor with entPhysicalClass of sensor(8) and discrete(3).	read	sunPlatDiscreteSensorEntry
Display the current reading of the sensor.	read	sunPlatDiscreteSensorCurrent

Related Information

- "OID Tables Overview" on page 153
- "SUN-PLATFORM-MIB MIB OID Command Syntax" on page 182

Discrete Sensor States Table OIDs

The following table lists discrete sensor state object identifiers supported by the SUN-PLATFORM-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display a table of available states for a discrete sensor of the Discrete Sensor Table.	read	sunPlatDiscreteSensorStatesTable
Display information about a sensor state for a sensor in the Discrete Sensor Table.	read	sunPlatDiscreteSensorStatesEntry
Display a number identifying the sensor state.	read	sunPlatDiscreteSensorStatesIndex
Display an interpretation of a discrete sensor state.	read	sunPlatDiscreteSensorStatesInterpretation
Display or set whether the state value is acceptable.	read	sunPlatDiscreteSensorStatesAcceptable

Related Information

- "OID Tables Overview" on page 153
- "SUN-PLATFORM-MIB MIB OID Command Syntax" on page 182

Sun Platform Fan Table OIDs

The following table lists fan object identifiers supported by the SUN-PLATFORM-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display a table of attributes common to all fans and cooling devices.	read	sunPlatFanTable
Display information about a piece of equipment with an entPhysicalClass of fan(7).	read	sunPlatFanEntry
Display the class of the cooling device.	read	sunPlatFanClass

Related Information

- "OID Tables Overview" on page 153
- "SUN-PLATFORM-MIB MIB OID Command Syntax" on page 182

Sun Platform Alarm Table OIDs

The following table lists alarm object identifiers supported by the SUN-PLATFORM-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display a table with indications of alarm states.	read	sunPlatAlarmTable
Display information about a piece of equipment with an sunPlatPhysicalClass of alarm(2).	read	sunPlatAlarmEntry
Display the type of alarm.	read	sunPlatAlarmType
Display or set the state of the alarm.	read write	sunPlatAlarmState
Display or set the urgency of the alarm.	read	sunPlatAlarmUrgency

Related Information

- "OID Tables Overview" on page 153
- "SUN-PLATFORM-MIB MIB OID Command Syntax" on page 182

MIB Trap OIDs

The following table lists MIB trap object identifiers supported by the SUN-PLATFORM-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display or set a unique ID for the notification.	read	sunPlatNotificationEventId
Display the time that the notification was generated.	read	sunPlatNotificationTime
Display the name of the object to which the notification applies.	read	sunPlatNotificationObject
Display the perceived severity of the alarm, as specified by the agent that generated it.	read	sunPlatNotificationPerceivedSeverity
Display or set an optional probable cause which provides additional information relevant to the notification.	read	sunPlatNotificationProbableCause

Description of Task or Activity	Action	MIB OID
Display or set an optional description which provides additional information concerning the cause of the notification.	read	sunPlatNotificationSpecificProblem
Display or set a list of recommended repair actions.	read	sunPlatNotificationRepairAction
Display an optional object identifier which provides additional information relevant to the notification.	read	sunPlatNotificationAdditionalInfo
Display or set an optional description which provides additional information relevant to the notification.	read	sunPlatNotificationAdditionalText
Display the changed object's OBJECT IDENTIFIER value in an attribute or state change notification.	read	sunPlatNotificationChangedobject identifier
Display the new INTEGER value in an attribute or state change notification.	read	sunPlatNotificationNewInteger
Display the old INTEGER value in an attribute or state change notification.	read	sunPlatNotificationOldInteger
Display the new OCTET-STRING value in an attribute change notification.	read	sunPlatNotificationNewString
Display the old OCTET-STRING value in an attribute change notification.	read	sunPlatNotificationOldString
Display the new OBJECT IDENTIFIER value in an attribute change notification.	read	sunPlatNotificationNewobject identifier
Display the old OBJECT IDENTIFIER value in an attribute change notification.	read	sunPlatNotificationOldobject identifier
Display or set a comma-delimited list of sunPlatNotificationEventIDs which are correlated to the notification.	read	sunPlatNotificationCorrelatedNotifications

Related Information

- "OID Tables Overview" on page 153
- "SUN-PLATFORM-MIB MIB OID Command Syntax" on page 182

Understanding the ENTITY-MIB MIB OIDs

This topic provides tables of physical and logical entity object identifiers.

Topics include:

Related Information

- "OID Tables Overview" on page 153
- "Understanding the SUN-DCS-MIB MIB OIDs" on page 154
- "SUN-HW-TRAP-MIB MIB OIDs" on page 170
- "Understanding the SUN-ILOM-CONTROL-MIB MIB OIDs" on page 171
- "Understanding the SUN-PLATFORM-MIB MIB OIDs" on page 181

ENTITY-MIB MIB OID Command Syntax

When you use the object identifiers in a command syntax, you must include the ENTITY-MIB MIB name. For example, using the V2c protocol:

```
$ command -v2c -c public mc_IP ENTITY-MIB::object_id argument
```

Or using the V3 protocol:

```
$ command -v3 -u snmp_user -l security_level -a authentication_protocol -A authentication_password
-x DES -X privacy_password mc_IP ENTITY-MIB::object_id argument
```

where:

- *command* is the command of the SNMP client that performs the action.
- *snmp_user* is a configured user of the SNMP services.
- *security_level* is:
  - noAuthNoPriv – There is no authentication or privacy.
  - authNoPriv – There is authentication, but no privacy.
  - authPriv – There is authentication and privacy.

- *authentication_protocol* is either MD5 or SHA
- *authentication_password* is the *snmp_user*'s authentication password.
- *privacy_password* is the *snmp_user*'s privacy password.
- *mc_IP* is the IP address of the management controller.
- *object_id* is the object identifier, as listed in the right column of the tables.
- *argument* is a combination of options and variables that support the object identifier.

For example, to securely display the discription of physical entity 4 on the management controller with IP address of 123.45.67.89:

```
$ snmpget -v3 -u usersnmp -l authPriv -a MD5 -A authpass -x DES -X privpass
123.45.67.89 ENTITY-MIB::entPhysicalDescr.4
```

Related Information

- "SUN-DCS-MIB MIB OID Command Syntax" on page 155
- "SUN-ILOM-CONTROL-MIB MIB OID Command Syntax" on page 172
- "SUN-PLATFORM-MIB MIB OID Command Syntax" on page 182

Physical Entity Table OIDs

The following table lists physical entity object identifiers supported by the ENTITY-MIB MIB.

Description of Task or Activity	Action	MIB OID
Display a table of information about the physical entities.	read	entPhysicalTable
Display information about a particular physical entity.	read	entPhysicalEntry
Display the index of the physical entity.	read	entPhysicalIndex
Display or set the description of the physical entity.	read	entPhysicalDescr
Display or set the vendor-specific hardware type of the physical entity.	read	entPhysicalVendorType
Display the value of entPhysicalIndex for the physical entity which is the container for this physical entity.	read	entPhysicalContainedIn

Description of Task or Activity	Action	MIB OID
Display or set the general hardware type of the physical entity.	read	entPhysicalClass
Display or set an indication of the relative position of the child component among all its sibling components.	read	entPhysicalParentRelPos
Display or set the name of the physical entity.	read	entPhysicalName
Display or set the vendor-specific hardware revision string of the physical entity.	read	entPhysicalHardwareRev
Display the vendor-specific firmware revision string of the physical entity.	read	entPhysicalFirmwareRev
Display the vendor-specific software revision string of the physical entity.	read	entPhysicalSoftwareRev
Display or set the vendor-specific serial number string for the physical entity.	read write	entPhysicalSerialNum
Display or set the name of the manufacturer of the physical component.	read	entPhysicalMfgName
Display or set the vendor-specific model name string associated with the physical component.	read	entPhysicalModelName
Display or set a handle or alias for the physical entity as specified by the network manager.	read write	entPhysicalAlias
Display or set the nonvolatile asset tracking identifier for the physical entity as specified by a network manager.	read write	entPhysicalAssetID
Display whether or not the physical entity is considered a field replaceable unit by the vendor.	read	entPhysicalIsFRU

Related Information

- "OID Tables Overview" on page 153
- "ENTITY-MIB MIB OID Command Syntax" on page 192

Index

C

D

E

G

H

I

L

M

O

P

S

U

V