

**Oracle® Server Management Agent for
Oracle Solaris 11.2 User's Guide**

ORACLE®

Part No: E51811-05
June 2015

Part No: E51811-05

Copyright © 2014, 2015, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS. Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Référence: E51811-05

Copyright © 2014, 2015, Oracle et/ou ses affiliés. Tous droits réservés.

Ce logiciel et la documentation qui l'accompagne sont protégés par les lois sur la propriété intellectuelle. Ils sont concédés sous licence et soumis à des restrictions d'utilisation et de divulgation. Sauf stipulation expresse de votre contrat de licence ou de la loi, vous ne pouvez pas copier, reproduire, traduire, diffuser, modifier, breveter, transmettre, distribuer, exposer, exécuter, publier ou afficher le logiciel, même partiellement, sous quelque forme et par quelque procédé que ce soit. Par ailleurs, il est interdit de procéder à toute ingénierie inverse du logiciel, de le désassembler ou de le décompiler, excepté à des fins d'interopérabilité avec des logiciels tiers ou tel que prescrit par la loi.

Les informations fournies dans ce document sont susceptibles de modification sans préavis. Par ailleurs, Oracle Corporation ne garantit pas qu'elles soient exemptes d'erreurs et vous invite, le cas échéant, à lui en faire part par écrit.

Si ce logiciel, ou la documentation qui l'accompagne, est concédé sous licence au Gouvernement des Etats-Unis, ou à toute entité qui délivre la licence de ce logiciel ou l'utilise pour le compte du Gouvernement des Etats-Unis, la notice suivante s'applique:

U.S. GOVERNMENT END USERS. Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

Ce logiciel ou matériel a été développé pour un usage général dans le cadre d'applications de gestion des informations. Ce logiciel ou matériel n'est pas conçu ni n'est destiné à être utilisé dans des applications à risque, notamment dans des applications pouvant causer des dommages corporels. Si vous utilisez ce logiciel ou matériel dans le cadre d'applications dangereuses, il est de votre responsabilité de prendre toutes les mesures de secours, de sauvegarde, de redondance et autres mesures nécessaires à son utilisation dans des conditions optimales de sécurité. Oracle Corporation et ses affiliés déclinent toute responsabilité quant aux dommages causés par l'utilisation de ce logiciel ou matériel pour ce type d'applications.

Oracle et Java sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut correspondre à des marques appartenant à d'autres propriétaires qu'Oracle.

Intel et Intel Xeon sont des marques ou des marques déposées d'Intel Corporation. Toutes les marques SPARC sont utilisées sous licence et sont des marques ou des marques déposées de SPARC International, Inc. AMD, Opteron, le logo AMD et le logo AMD Opteron sont des marques ou des marques déposées d'Advanced Micro Devices. UNIX est une marque déposée d'The Open Group.

Ce logiciel ou matériel et la documentation qui l'accompagne peuvent fournir des informations ou des liens donnant accès à des contenus, des produits et des services émanant de tiers. Oracle Corporation et ses affiliés déclinent toute responsabilité ou garantie expresse quant aux contenus, produits ou services émanant de tiers, sauf mention contraire stipulée dans un contrat entre vous et Oracle. En aucun cas, Oracle Corporation et ses affiliés ne sauraient être tenus pour responsables des pertes subies, des coûts occasionnés ou des dommages causés par l'accès à des contenus, produits ou services tiers, ou à leur utilisation, sauf mention contraire stipulée dans un contrat entre vous et Oracle.

Accessibilité de la documentation

Pour plus d'informations sur l'engagement d'Oracle pour l'accessibilité à la documentation, visitez le site Web Oracle Accessibility Program, à l'adresse <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Accès au support électronique

Les clients Oracle qui ont souscrit un contrat de support ont accès au support électronique via My Oracle Support. Pour plus d'informations, visitez le site <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> ou le site <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> si vous êtes malentendant.

Contents

Using This Documentation	9
Oracle Server Management Agent User's Guide Overview	13
About the Oracle Server Hardware Management Agent Components	15
Oracle Server Hardware Management Agent	15
Oracle Server Hardware SNMP Plugins	16
itpconfig and the ILOM Trap Proxy	16
Configuring Hardware Management Agent and Hardware SNMP Plugins	17
Hardware Management Agent Configuration File	17
Configuring the Hardware Management Agent Logging Level	17
▼ How to Configure the Hardware Management Agent Logging Level	19
Configuring Net-SNMP In Oracle Solaris	19
▼ How to Configure SNMP Gets	20
▼ How to Configure SNMP Sets	20
▼ How to Configure SNMP Traps	20
▼ How to Restart Net-SNMP	21
Oracle Server Hardware SNMP Plugins Overview	23
Overview of Sun HW Monitoring MIB	23
Sun Server Product and Chassis	24
Sun Server Service Processor	24
Sun Server Hardware Monitoring MIB	24
Sun Server Hardware Management Agent	25
Sun Server Hardware Inventory	25
Sun Server Hardware Monitor Sensor Group	25
sunHwMonIndicatorGroup	26
sunHwMonTotalPowerConsumption	27
Overview of Sun HW Trap MIB	27

Overview of Sun Storage MIB	28
Sun Storage MIB Objects	28
Physical and Logical Storage Objects	28
Working With the Hardware Management Agent	31
Retrieving and Setting Information Through SNMP	31
sunHwMonProductGroup	31
▼ How to Retrieve the Product Information from an Oracle x86 Server	32
▼ How to Retrieve The Product Information on an Oracle x86 Server Module	32
sunHwMonProductChassisGroup	33
▼ How to Retrieve the Server Module's Product Chassis Information	33
sunHwMonSPGroup	34
▼ How to Retrieve Service Processor Information	34
sunHwMonInventoryTable	34
▼ How to Retrieve Inventory Information	35
sunHwMonSensorGroup	35
▼ How to Retrieve the Sensor Group Information	35
sunHwMonIndicatorLocator	37
▼ How to Set the Indicator Locator	38
Generating SNMP Traps	38
▼ How to Inject a Simulated Fault	38
Using the itpconfig Tool	41
itpconfig Command Usage	41
Options	41
Subcommands	42
Error Codes	42
itpconfig Usage Scenario	42
Host-to-ILOM Interconnect Configuration Commands	43
▼ How to Enable Host-to-ILOM Interconnect	43
▼ How to Disable Host-to-ILOM Interconnect	44
▼ How to List the Host-to-ILOM Interconnect Settings	44
itpconfig Trap Forwarding Commands	44
▼ How to Enable Trap Forwarding	45
▼ How to Disable Trap Forwarding	45
▼ How to List Trap Forwarding Settings	45

Using Oracle Hardware Management Pack to Monitor Disk Diagnostic Events	47
Monitoring Disk Events	47
Troubleshooting the Hardware Management Agent	49
itpconfig Troubleshooting	49
The Hardware Management Agent Continually Goes Into Maintenance State	49
How Can I Determine if SNMP Is Configured Correctly for the Hardware Management Agent?	50
SNMP Data Is Not Available Immediately After Starting the Hardware Management Agent	50
Where Are the MIB Files Located?	51
Index	53

Using This Documentation

This section describes how to get the latest documentation and support for Oracle Hardware Management Pack (HMP) for Oracle Solaris. It also provides feedback links and a document change history.

- [“About Oracle Hardware Management Pack for Oracle Solaris” on page 9](#)
- [“Documentation and Feedback” on page 9](#)
- [“About This Documentation” on page 10](#)
- [“Support and Training” on page 10](#)
- [“Contributing Authors” on page 10](#)
- [“Change History” on page 10](#)

About Oracle Hardware Management Pack for Oracle Solaris

Oracle Hardware Management Pack for Oracle Solaris is a set of commands and agents that help you manage your Oracle servers and is part of the Oracle Solaris operating system starting with Oracle Solaris 11.2.

If you have an earlier version of Oracle Solaris, you can obtain the standalone version of Oracle Hardware Management Pack through the Oracle Support site.

Documentation and Feedback

The following documentation is available related to the Oracle Hardware Management Pack for Oracle Solaris.

Documentation	Link
All Oracle products	http://docs.oracle.com
Oracle Hardware Management Pack for Oracle Solaris	http://www.oracle.com/goto/ohmp/solarisdocs

Documentation	Link
Oracle ILOM	http://www.oracle.com/goto/ilom/docs

Provide feedback on this documentation at:

<http://www.oracle.com/goto/docfeedback>.

About This Documentation

This documentation is available in both PDF and HTML. The information is presented in topic-based format (similar to online help) and therefore does not include chapters, appendixes, or section numbering.

Support and Training

These web sites provide additional resources:

- Support: <https://support.oracle.com>
- Training: <http://education.oracle.com>

Contributing Authors

The following authors contributed to this documentation: Cynthia Chin-Lee, Lisa Kuder, David Moss, Ralph Woodley, Michael Bechler.

Change History

The following changes have been made to the document.

- July 2014. Initial publication.
- December 2014. Updated *Release Notes* to document bug 19462769.
- January 2015. Updated the *Management Agent User's Guide* to correct the name of the hardware management agent service for Solaris 11.2 and later.
- March 2015. Updated *CLI User's Guide* and *Installation Guide* to add the `nvmeadm` command. Updated *CLI User's Guide* to add error codes for the `ubiosconfig` command. Made general editorial improvements and other minor technical updates.

- June 2015. Updated the *Release Notes* to document bug 21098717. Updated *CLI User's Guide* to describe new raidconfig read/write cache options. Updated *Management Agent User's Guide* to add section on enhanced disk diagnostic events. Made general editorial improvements and other minor technical updates.

Oracle Server Management Agent User's Guide Overview

This guide provides an overview of Oracle Server Hardware Management Agent and how to use it with your Oracle server.

Note - This documentation applies to servers running Oracle Solaris 11.2 and later versions.

Beginning with Oracle Solaris 11.2, Oracle Hardware Management Pack (HMP) has become an integrated component of the operating system and is called Oracle HMP for Oracle Solaris. Do not download and use other versions of Oracle Hardware Management Pack that are not specifically qualified for the Oracle Solaris 11.2 (and later) operating system. If you have Oracle Solaris 11.1 or earlier or other operating systems, continue to use Oracle HMP, available as a separate download from <https://support.oracle.com>.



Oracle Hardware Management Pack (HMP) for Oracle Solaris is a set of commands and an agent to help you manage your Oracle servers. Now part of Oracle Solaris 11.2, this software was previously a separate download.

This software has features similar to Oracle HMP 2.2.8, which is available for servers with Oracle Solaris 11.1 and earlier and for servers with other operating systems.

- [“About the Oracle Server Hardware Management Agent Components”](#)
- [“Configuring Hardware Management Agent and Hardware SNMP Plugins”](#)
- [“Oracle Server Hardware SNMP Plugins Overview”](#)
- [“Working With the Hardware Management Agent”](#)
- [“Using the `itpconfig` Tool”](#)
- [“Troubleshooting the Hardware Management Agent”](#)

For information on installing the Hardware Management Agent, see [“Oracle Hardware Management Pack for Oracle Solaris 11.2 Installation Guide”](#).

About the Oracle Server Hardware Management Agent Components

Oracle Hardware Management Pack includes a Hardware Management Agent to help you monitor Oracle servers.

The Hardware Management Agent includes the following components:

- [“Oracle Server Hardware Management Agent” on page 15](#)
- [“Oracle Server Hardware SNMP Plugins” on page 16](#)
- [“itpconfig and the ILOM Trap Proxy” on page 16](#)

Oracle Server Hardware Management Agent

The Oracle Server Hardware Management Agent (Hardware Management Agent) and associated Oracle Server Hardware SNMP Plugins (Hardware SNMP Plugins) provide a way to monitor and manage your server hardware using an operating system native agent. This in-band functionality enables you to use a single IP address (the host's IP) for monitoring your servers and blade server modules without having to connect the management port of the Oracle Integrated Lights Out Manager (ILOM) service processor to the network.

The Hardware Management Agent and Hardware SNMP Plugins run on the host operating system of your Oracle servers, communicating with the Oracle ILOM service processor. The Hardware Management Agent service, called `svc:/system/sp/management:default` (called `hwmgmtd` in Solaris versions earlier than 11.2), regularly polls the service processor for information about the current state of the server. Hardware Management Agent can poll the service processor for hardware information over either the Host-to-ILOM Interconnect, available on Oracle latest servers, or KCS interface on previous generation servers. This information is then made available by Hardware Management Agent over SNMP using the Hardware SNMP Plugins.

In addition, the Hardware Management Agent also maintains a separate log that contains information about the Hardware Management Agent status, which can be used for troubleshooting.

Oracle Server Hardware SNMP Plugins

The Oracle Server Hardware SNMP Plugins consists of Net-SNMP plugins, that are compiled versions of hardware-specific Management Information Bases (MIB) which have been designed to enable you to monitor your Oracle servers effectively.

The sunHwMonMIB describes the state of sensors and alarms on your servers and provides the following information:

- Overall system alarm status
- Aggregate alarm status by device type
- FRU Alarm status
- Lists of sensors, sensor types, sensor readings, and sensor thresholds
- Indicator states
- System locator control
- Inventory including basic manufacturing information
- Product and chassis inventory information (such as serial number and part numbers)
- Per-sensor alarm status

The sunHwTrapMIB describes a set of traps for hardware events that can be generated by an Oracle server and provides the following information:

- Conditions affecting the environmental state of the server (such as temperature, voltage, and current out-of-range conditions)
- Error conditions affecting the hardware components in the server such as FRU insertion and removal and security intrusion notification

The sunStorageMIB provides the following information about system storage:

- Basic manufacturing information, properties, and alarm status for controllers
- Properties and alarm status for disks
- Properties and alarm status for RAID volumes
- Status of logical components

itpconfig and the ILOM Trap Proxy

The `itpconfig` tool enables you to configure a trap proxy on the host to forward SNMP traps generated from Oracle ILOM based service processors over the Host-to-ILOM Interconnect to a configurable destination. `itpconfig` can also enable or disable the Host-to-ILOM Interconnect, which is available on the latest Oracle servers.

Configuring Hardware Management Agent and Hardware SNMP Plugins

This section provides instructions about configuring the Hardware Management Agent and Hardware SNMP Plugins, as well as information about using Hardware Management Agent.

- [“Hardware Management Agent Configuration File” on page 17](#)
- [“Configuring the Hardware Management Agent Logging Level” on page 17](#)
- [“How to Configure the Hardware Management Agent Logging Level” on page 19](#)
- [“Configuring Net-SNMP In Oracle Solaris” on page 19](#)

Hardware Management Agent Configuration File

Once the Hardware Management Agent and Hardware SNMP Plugins are installed on the Oracle server you want to monitor, you can configure the level of detail used for log messages using the `hwmgmt.d.conf` file.

The Hardware Management Agent records log messages into the log file. These messages can be used to troubleshoot the running status of the Hardware Management Agent. The Hardware Management Agent records the log messages used for troubleshooting in the following log file:

```
/var/log/ssm/hwmgmt.d.log
```

The level of detail of the messages recorded in the log file depends on the logging level set in the configuration file.

Configuring the Hardware Management Agent Logging Level

To configure the logging level, modify the `hwagentd_log_levels` parameter in the `hwmgmt.d.conf` file. There are two ways to configure the logging level. The easiest way to configure the logging level is to set the `hwagentd_log_levels` parameter to one of the following levels.

Log Level	Messages Logged
ERROR	Any error messages generated by the Hardware Management Agent
WARNING	Any error and warning messages generated by the Hardware Management Agent
INFO	Any error and warning messages generated by the Hardware Management Agent and informative messages about normal functioning

Alternatively, you can set the logging level with a finer level of granularity by using the bit flags from the following table.

Note - It is recommended to use the logging levels above. The following options are for advanced troubleshooting.

Log Level	Bit Code	Messages Logged
EMERG	0x0001	Information about the system being unusable
ALARM	0x0002	Information about any immediate action that must be taken
CRIT	0x0004	Information related to the Hardware Management Agent either not starting or stopping because of critical conditions
ERROR	0x0008	Information about any error messages generated by the Hardware Management Agent
WARNING	0x0010	Information about any error and warning messages generated by the Hardware Management Agent
NOTICE	0x0020	Information related to normal functioning
INFO	0x0040	Information about any error and warning messages generated by the Hardware Management Agent and informative messages about normal functioning
DEBUG	0x0080	Verbose debug-level messages, useful in troubleshooting
TRACE	0x0100	Highly verbose debug-level messages, useful in troubleshooting

Note - levels DEBUG and TRACE generate a lot of detailed messages and are designed for troubleshooting. These levels are not recommended for production usage.

For example, when you want to set all logging levels between EMERG and NOTICE, the bit code values of all the required levels must be added and then converted to a decimal value. Referring to preceding table, the addition would be as follows:

$$0x0001 + 0x0002 + 0x0004 + 0x0008 + 0x0010 + 0x0020 = 0x003f$$

Converting this hexadecimal value to decimal equals 63, which is the desired log level. This is the decimal number that should be assigned to the `hwagentd_log_levels` parameter in the `hwmgmtd.conf` file.

▼ How to Configure the Hardware Management Agent Logging Level

1. Find the `hwmgmt.d.conf` file and open it for editing.

The file is located:

```
/etc/ssm/hwmgmt.d.conf
```

2. Find the `hwagentd_log_levels` parameter and change the logging level to one of the options from the tables above.

3. Save the modified `hwmgmt.d.conf` file.

4. Disable and re-enable the Hardware Management Agent, which forces the `hwmgmt.d.conf` to be reread. Enter the following commands:

```
svcadm disable svc:/system/sp/management:default
```

```
svcadm enable svc:/system/sp/management:default
```

The Hardware Management Agent rereads the `hwmgmt.d.conf` file with the modified `hwagentd_log_levels` parameter.

Configuring Net-SNMP In Oracle Solaris

The Hardware Management Agent uses SNMP for network communications. For the Hardware Management Agent to be able to use SNMP correctly on host operating systems, you must ensure that SNMP is configured correctly. Incorrect settings can cause the Hardware Management Agent to have limited, or no network connectivity.

In Oracle Solaris operating system, the `snmpd.conf` file controls network access to the Hardware Management Agent. You can find the `snmpd.conf` file in the following location:

```
/etc/net-snmplib/snmp/snmpd.conf
```

The following procedures explain how to configure SNMP gets, sets, and traps.

Note - The following instructions assume you are using an unmodified `snmpd.conf` file. If you have customized your `snmpd.conf` file, use these instructions as a guide to make sure your `snmpd.conf` file is compatible with the Hardware Management Agent.

This section covers the following procedures:

- [“How to Configure SNMP Gets” on page 20](#)

- [“How to Configure SNMP Sets” on page 20](#)
- [“How to Configure SNMP Traps” on page 20](#)
- [“How to Restart Net-SNMP” on page 21](#)

▼ How to Configure SNMP Gets

SNMP gets enable you to read data filled by the Hardware Management Agent. To be able to perform SNMP gets, use the following information to modify your `snmpd.conf` file.

1. **Open your `snmpd.conf` file for editing.**
2. **For Oracle Solaris operating system, add the following line to `snmpd.conf`:**

```
rocommunity public
```

This adds a read-only community from a network location other than localhost.

▼ How to Configure SNMP Sets

To enable the functionality of setting information over SNMP, use the following information to modify your `snmpd.conf` file.

1. **Open your `snmpd.conf` file for editing.**
2. **For Oracle Solaris operating system, add the following line:**

```
rwcommunity private
```

By default the public community is blocked as rocommunity on Oracle Solaris.

▼ How to Configure SNMP Traps

1. **Open your `snmpd.conf` file for editing.**
2. **Depending on the version of SNMP traps you want to send:**
 - **To be able to send SNMP version 1 traps from the Hardware Management Agent, add the following line to `snmpd.conf`:**

```
trapsink host communitystring trapport
```

- **To be able to send SNMP version 2 traps from the Hardware Management Agent, add the following line to `snmpd.conf`:**

```
trap2sink host communitystring trapport
```

Example 1 Setting SNMP Version 2 Traps

The following example shows the line added to the `snmpd.conf` file to configure SNMP Traps using SNMP version 2:

```
trap2sink 10.18.141.22 public 162
```

▼ How to Restart Net-SNMP

- **If Oracle Hardware SNMP Plugins are installed, restart the SNMP daemon using the following command:**

```
svcadm restart svc:/application/management/net-snmp:default
```


Oracle Server Hardware SNMP Plugins Overview

This section contains overviews of the Management Information Bases (MIBs) that are implemented by Oracle Server Hardware SNMP Plugins. If the `system/management/hmp-snmp` package is installed, Hardware Management Pack MIBs are located in:

`/usr/lib/asm/lib/mibs`

- [“Overview of Sun HW Monitoring MIB” on page 23](#)
- [“Overview of Sun HW Trap MIB” on page 27](#)
- [“Overview of Sun Storage MIB” on page 28](#)

Overview of Sun HW Monitoring MIB

The Sun HW Monitoring Management Information Base (MIB) provides the following details about the server or server module implementing this MIB:

- A hardware inventory of all Field Replaceable Units (FRU) and sensors monitoring different physical parameters
- Parent/child relationship or containment information of all FRUs and sensors
- Individual status of each sensor as well as combined status of each device type
- Any threshold values configured for each sensor, where applicable
- Details about the service processor
- Information about total power consumption

The MIB is subdivided into sections, based on the information provided by the MIB objects. The information provided by the MIB objects is categorized into logically divided groups of scalars, as well as MIB tables.

For a complete list of all of the objects defined by each group, refer to the comments section defined at the beginning of each group in the `SUN-HW-MONITORING-MIB.mib` file.

The following sections briefly describe each of the MIB sections, with some examples of the objects defined in each group:

- [“Sun Server Product and Chassis” on page 24](#)
- [“Sun Server Service Processor” on page 24](#)
- [“Sun Server Hardware Monitoring MIB” on page 24](#)
- [“Sun Server Hardware Management Agent” on page 25](#)
- [“Sun Server Hardware Inventory” on page 25](#)
- [“Sun Server Hardware Monitor Sensor Group” on page 25](#)
- [“sunHwMonIndicatorGroup” on page 26](#)
- [“sunHwMonTotalPowerConsumption” on page 27](#)

Sun Server Product and Chassis

The first two groups, sunHwMonProductGroup and sunHwMonProductChassisGroup, define scalar MIB objects that provide information about the server, including part number, and manufacturer. These groups are:

- sunHwMonProductGroup is a scalar group that provides general product details about the server or server module, such as the part number, type, name, and serial number.
- sunHwMonProductChassisGroup is a scalar group that provides details about the server's chassis or the chassis into which the server has been inserted.

Note - sunHwMonProductChassisGroup is populated only on server modules, where it is relevant.

Sun Server Service Processor

The Sun Server Service Processor group consists of one group, sunHwMonSPGroup, which is a scalar group that provides details about the server's Oracle Integrated Lights Out Management (ILOM) service processor. This group includes information such as serial number, manufacturer, MAC Address, IP details, and Web accessibility information such as the URL to access the Oracle ILOM Web interface.

Sun Server Hardware Monitoring MIB

The Sun Server Hardware Monitoring MIB group consists of one scalar group, sunHwMonMibGroup that provides details about the SUN-HW-MONITORING-MIB itself, such as MIB version number.

Sun Server Hardware Management Agent

The Sun Servers Hardware Management Agent group consists of one scalar group, `sunHwMonAgentSoftwareGroup` that provides details about the Hardware Management Agent associated with this MIB, such as the version of the agent and the connection status to Oracle ILOM.

Sun Server Hardware Inventory

The Sun Servers Hardware Inventory group consists of one scalar group, `sunHwMonInventoryGroup` with a MIB table, `sunHwMonInventoryTable`. This table contains details about the server's field replaceable units (FRUs). For each FRU, it includes the name, type, description, part number, status, and the FRU in which it is contained (if any).

Sun Server Hardware Monitor Sensor Group

The `sunHwMonSensorGroup` contains details about all of the server's hardware sensors, except indicators. The MIB objects that define the sensor properties are hierarchically and logically grouped based on device type, for example temperature or voltage, as well as sensor type, for example numeric or discrete.

The `sunHwMonSensorGroup` also contains a device-specific group for all significant device types, such as `sunHwMonVoltageGroup` or `sunHwMonCurrentGroup`. There is also a group for sensors that are not part of any device—specific group.

Each of the groups listed below contains two tables. One table provides details about all of the numeric sensors of this device type and the other table provides details about all of the discrete sensors of corresponding device type on the server.

The numeric sensors tables provide details about numeric sensors such as the sensor name, sensor type, the current reading, defined thresholds, current status, perceived severity, and the FRU in which the sensor is contained. The discrete sensors tables provide details about discrete sensors, such as sensor name, sensor type, sensor state, perceived severity, and the FRU in which the sensor is contained.

The alarm status of an entity can be one of the following, where critical is the most severe and indeterminate is the least severe.

- critical
- major
- minor
- warning

- cleared
- indeterminate

The sunHwMonSensorGroup contains the following subgroups:

sunHwMonSensorGroup Groups	Description
sunHwMonSensorAlarmStatusGroup	This is a scalar group that provides a single view of the alarm status of the server and aggregate status per device type such as rolled-up status of all voltage sensors. This is the main value used to obtain the overall status of a server. The individual sensor status is provided by MIB objects that are defined in the corresponding device-specific group.
sunHwMonVoltageGroup	Contains two MIB tables that provide details regarding all voltage sensors contained in the server.
sunHwMonCurrentGroup	Contains two MIB tables that provide details regarding all current sensors contained in the server.
sunHwMonPowerDeviceGroup	Contains two MIB tables that provide details regarding all power device sensors contained in the server.
sunHwMonCoolingDeviceGroup	Contains two MIB tables that provide details regarding all cooling device sensors contained in the server.
sunHwMonTemperatureGroup	Contains two MIB tables that provide details regarding all temperature sensors contained in the server.
sunHwMonMemoryGroup	Contains two MIB tables that provide details regarding all memory sensors contained in the server.
SunHwMonProcessorGroup	Contains two MIB tables that provide details regarding all processor sensors contained in the server.
sunHwMonHardDriveGroup	Contains two MIB tables that provide details regarding all hard drive sensors contained in the server.
sunHwMonIOGroup	Contains two MIB tables that provide details regarding all input/output sensors contained in the server.
sunHwMonSlotOrConnectorGroup	Contains two MIB tables that provide details regarding all slot or connector sensors contained in the server.
sunHwMonOtherSensorGroup	Contains two MIB tables that provide details regarding all sensors contained in the server that are not part of above defined device type groups.

sunHwMonIndicatorGroup

This group contains multiple groups that provide details about the indicators present on the server. These groups are as follows:

sunHwMonIndicatorGroup Groups	Description
sunHwMonIndicatorLocator	This is a scalar group that provides details about the locator indicator, such as the name of the locator

sunHwMonIndicatorGroup Groups	Description
	indicator sensor and its status. It contains the sunHwMonIndicatorLocatorCurrentStatus MIB object, which is a read-write MIB object. You can control the locator indicator sensor through an SNMP set command, using a community string with write access.
sunHwMonIndicatorService	This is a scalar group that provides the name and status of the service indicator sensor.
sunHwMonIndicatorAll	This group contains sunHwMonIndicatorTable, which provides details about all indicators present on the server, such as power supply failure indicator or fan failure indicator.

sunHwMonTotalPowerConsumption

This scalar group provides details about the server's total power consumption, including:

- Sensor name and type
- Current reading
- Defined thresholds
- Current status
- Perceived severity
- The FRU in which the sensor is contained

Note - Data is available here only if the platform has implemented a total power consumption indicator.

Overview of Sun HW Trap MIB

The Hardware Management Agent uses the Sun HW Trap MIB to implement SNMP traps. These traps report the environmental state of the server as well as faults, errors, and other conditions affecting hardware components.

The SNMP traps are categorized into three groups.

- Any SNMP trap name ending in Ok or Error, as well as any SNMP trap name containing Threshold, is reporting a change in a sensor value.
- Any SNMP trap name ending in Fault is reporting a problem detected by the system's fault management subsystem, if such a subsystem is available on the server.
- The final group is the status SNMP traps, which report the environmental state and any hardware information that is not covered by the two previous groups.

For more detailed information on the Sun HW Trap MIB, see the comments in the SUN-HW-TRAP-MIB.mib file.

Overview of Sun Storage MIB

The Sun Storage MIB supplements the Sun HW Monitoring MIB with storage-related information. The following sections briefly describe each of the MIB sections:

- [“Sun Storage MIB Objects” on page 28](#)
- [“Physical and Logical Storage Objects” on page 28](#)

Sun Storage MIB Objects

The following scalar objects contain information about the Sun Storage MIB itself:

- `sunStorageAgentVersion` defines the version of the software implementing the `sunStorageMIB`. The version is in the following format: *MajorVersion.MinorVersion.SubMinorVersion* (for example: 1.2.3).
- `sunStorageMibVersion` defines the version of the SUN-STORAGE-MIB this agent implements. The version defined is in the format of *MajorVersion.MinorVersion.SubMinorVersion* (for example: 1.3.0).

Physical and Logical Storage Objects

The following tables list physical and logical storage objects:

- `sunStorageControllerTable`. The storage controller object represents either an on-board or bus-attached storage controller. The properties associated with a controller object describe the type of controller (vendor and model) as well as the features it supports (such as RAID). The table is indexed with an arbitrary integer to uniquely identify each entry. Entries can contain the following:
 - Identifying: name, part number, serial number, manufacturer, model, firmware version, and PCIbus address
 - RAID capabilities: levels supported, maximum volumes manageable, number of spares, and stripe size
 - Status: operational and alarm
- `sunStorageDiskTable`. Each disk object corresponds to one physical disk that is available to the host operating system. Entries in this table might have parent objects in other tables (such as `sunStorageControllerTable`). The table is indexed with `sunHwMonFruIndex`, so that information corresponding to the same physical disk is retrievable from both the `sunHwMonInventoryTable` and `sunStorageDiskTable` at the same index.

- Identifying: name and OS device name
- Relational: parent name and index, slot number
- Descriptive: physical type, interface type, and capacity
- Status: mapping, RAID, and operational
- Entries can contain the following:
- sunStorageVolumeTable. This table contains logical volume objects that correspond to a logical disk visible to the host OS. Only RAID logical volumes are supported. The table is indexed with an arbitrary integer to uniquely identify each entry. Entries can contain the following:
 - Identifying: name, OS device name, and mount point
 - Relational: parent name and index
 - Descriptive: capacity, RAID level, and sizing
 - Status: mapping, mounting, RAID parameters, task, and operational
- sunStorageLogicalCompTable. A logical component node represents an active or passive component of its logical device parent. A logical component object is always a direct child of a logical device node. In the case of a RAID logical device, the logical component represents a physical device, or part of a physical device, used to create the specified RAID level. The table is indexed with an arbitrary integer to uniquely identify each entry. Entries can contain the following:
 - Identifying: name, disk name, and index
 - Relational: parent name and index
 - Status: RAID spare and RAID operational

Working With the Hardware Management Agent

Once the Hardware Management Agent is installed, you can use it to monitor your Oracle server. The Hardware Management Agent provides the SNMP Plugins layer, which enables you to retrieve and set information using SNMP, and to generate SNMP traps.

- [“Retrieving and Setting Information Through SNMP” on page 31](#)
- [“sunHwMonProductGroup” on page 31](#)
- [“sunHwMonProductChassisGroup” on page 33](#)
- [“sunHwMonSPGroup” on page 34](#)
- [“sunHwMonInventoryTable” on page 34](#)
- [“sunHwMonSensorGroup” on page 35](#)
- [“sunHwMonIndicatorLocator” on page 37](#)
- [“Generating SNMP Traps” on page 38](#)

Retrieving and Setting Information Through SNMP

The following section provides some examples of using Net-SNMP's `snmpwalk` utility to get and set information from Oracle servers running the Hardware Management Agent. For more information on the Hardware Management Agent functionality shown here, see [“Overview of Sun HW Monitoring MIB” on page 23](#) or the `SUN-HW-MONITORING-MIB.mib` file.

The format of the Net-SNMP `snmpwalk` command is:

```
snmpwalk Application options Common Options OID
```

For more information, see the Net-SNMP documentation.

sunHwMonProductGroup

The `sunHwMonProductGroup` contains information about the server implementing the MIB.

The following procedures are covered in this section:

- [“How to Retrieve the Product Information from an Oracle x86 Server” on page 32](#)

- [“How to Retrieve The Product Information on an Oracle x86 Server Module” on page 32](#)

▼ How to Retrieve the Product Information from an Oracle x86 Server

- **At the command prompt, type the following:**

```
# snmpwalk -v2c -c public -mALL localhost\ SUN-HW-MONITORING-MIB::  
sunHwMonProductGroup
```

You should see output similar to the following:

```
SUN-HW-MONITORING-MIB::sunHwMonProductName.0 = STRING: SUN FIRE X4440  
SUN-HW-MONITORING-MIB::sunHwMonProductType.0 = INTEGER: rackmount(3)  
SUN-HW-MONITORING-MIB::sunHwMonProductPartNumber.0 = STRING: 602-4058-01  
SUN-HW-MONITORING-MIB::sunHwMonProductSerialNumber.0 = STRING: 0823QBU01C  
SUN-HW-MONITORING-MIB::sunHwMonProductManufacturer.0 = STRING: SUN MICROSYSTEMS  
SUN-HW-MONITORING-MIB::sunHwMonProductSlotNumber.0 = INTEGER: -1  
  
SUN-HW-MONITORING-MIB::sunHwMonProductUUID.0 = STRING:  
080020FFFFFFFFFFFFFF00144FEDE5E0  
  
SUN-HW-MONITORING-MIB::sunHwMonProductBiosVersion.0 = STRING: S90_3B18
```

Note - On an Oracle x86 rack mount server, the following line signifies that there is no slot number (nodef).

```
sunHwMonProductSlotNumber.0 = INTEGER: -1
```

This is expected behavior because slot numbers are relevant only to blade servers. Rackmount servers do not have slot numbers.

▼ How to Retrieve The Product Information on an Oracle x86 Server Module

- **At the command prompt, type the following:**

```
# snmpwalk -v2c -c public -mALL localhost\ SUN-HW-MONITORING-MIB::  
sunHwMonProductGroup
```

You should see output similar to the following:

```
SUN-HW-MONITORING-MIB::sunHwMonProductName.0 = STRING: Sun Blade X6250 Server
Module
```

```
SUN-HW-MONITORING-MIB::sunHwMonProductType.0 = INTEGER: blade(4)
```

```
SUN-HW-MONITORING-MIB::sunHwMonProductPartNumber.0 = STRING: 540-7254-01
```

```
SUN-HW-MONITORING-MIB::sunHwMonProductSerialNumber.0 = STRING: 142300943223
```

```
SUN-HW-MONITORING-MIB::sunHwMonProductManufacturer.0 = STRING: Sun Microsystems
Inc
```

```
SUN-HW-MONITORING-MIB::sunHwMonProductSlotNumber.0 = INTEGER: 1
```

```
SUN-HW-MONITORING-MIB::sunHwMonProductUUID.0 = STRING:
080020FFFFFFFFFFFFFFFF01B24782F9C
```

```
SUN-HW-MONITORING-MIB::sunHwMonProductBiosVersion.0 = STRING: S90_3B18
```

sunHwMonProductChassisGroup

This group is filled only on Sun x86 server modules and represents the chassis holding the server module.

▼ How to Retrieve the Server Module's Product Chassis Information

- **At the command prompt, type the following:**

```
# snmpwalk -v2c -c public -mALL localhost\ SUN-HW-MONITORING-MIB::
sunHwMonProductChassisGroup
```

You should see output similar to the following:

```
SUN-HW-MONITORING-MIB::sunHwMonProductChassisName.0 = STRING: SUN BLADE 6000
MODULAR SYSTEM
```

```
SUN-HW-MONITORING-MIB::sunHwMonProductChassisPartNumber.0 = STRING: 541-1983-07
```

```
SUN-HW-MONITORING-MIB::sunHwMonProductChassisSerialNumber.0 = STRING: 1005LCB-
0728YM01R7
```

```
SUN-HW-MONITORING-MIB::sunHwMonProductChassisManufacturer.0 = STRING: SUN
MICROSYSTEMS
```

sunHwMonSPGroup

This group contains information about the Oracle ILOM service processor.

▼ How to Retrieve Service Processor Information

- **At the command prompt, type the following:**

```
# snmpwalk -v2c -c public -mALL localhost\ SUN-HW-MONITORING-MIB::  
sunHwMonSPGroup
```

You should see output similar to the following:

```
SUN-HW-MONITORING-MIB::sunHwMonSPSerialNumber.0 = STRING: 1762TH1-0750000707  
SUN-HW-MONITORING-MIB::sunHwMonSPManufacturer.0 = STRING: ASPEED  
SUN-HW-MONITORING-MIB::sunHwMonSPFWVersion.0 = STRING: 2.0.3.10  
SUN-HW-MONITORING-MIB::sunHwMonSPMacAddress.0 = STRING: 0:1b:24:78:2f:a1  
SUN-HW-MONITORING-MIB::sunHwMonSPIPAddress.0 = IPAddress: 10.18.141.164  
SUN-HW-MONITORING-MIB::sunHwMonSPNetMask.0 = IPAddress: 255.255.255.128  
SUN-HW-MONITORING-MIB::sunHwMonSPDefaultGateway.0 = IPAddress: 10.18.141.129  
SUN-HW-MONITORING-MIB::sunHwMonSPIPMode.0 = INTEGER: dhcp(2)  
SUN-HW-MONITORING-MIB::sunHwMonSPURLToLaunch.0 = STRING:  
SUN-HW-MONITORING-MIB::sunHwMonSPSystemIdentifier.0 = STRING:
```

Note - If the server uses Oracle ILOM 2.0, the following lines are returned:

```
SUN-HW-MONITORING-MIB::sunHwMonSPURLToLaunch.0 = STRING:  
SUN-HW-MONITORING-MIB::sunHwMonSPSystemIdentifier.0 = STRING:
```

This is expected behavior because this information is specific to Oracle ILOM 3.0.

sunHwMonInventoryTable

Information about only one FRU, `mb.net0.fru`, is shown in this example.

▼ How to Retrieve Inventory Information

- **At the command prompt, type the following:**

```
# snmpwalk -v2c -c public -mALL localhost\ SUN-HW-MONITORING-MIB::
sunHwMonInventoryTable | grep '.148 = '
```

where `grep '.148 = '` is filtering for results with a property of the FRU we are interested in. You should see output similar to the following:

```
SUN-HW-MONITORING-MIB::sunHwMonFruName.148 = STRING: /SYS/MB/NET0
SUN-HW-MONITORING-MIB::sunHwMonFruType.148 = INTEGER: networkInterface(80)
SUN-HW-MONITORING-MIB::sunHwMonFruDescr.148 = STRING:
SUN-HW-MONITORING-MIB::sunHwMonFruPartNumber.148 = STRING: 82546GB
SUN-HW-MONITORING-MIB::sunHwMonFruSerialNumber.148 = STRING: 00:14:4F:A8:39:44
SUN-HW-MONITORING-MIB::sunHwMonFruManufacturer.148 = STRING:
SUN-HW-MONITORING-MIB::sunHwMonFruStatus.148 = INTEGER: indeterminate(6)
SUN-HW-MONITORING-MIB::sunHwMonParentFruIndex.148 = INTEGER: 146
SUN-HW-MONITORING-MIB::sunHwMonParentFruName.148 = STRING: /SYS/MB
```

Note - If the server uses Oracle ILOM 2.0, the following lines are returned:

```
SUN-HW-MONITORING-MIB::sunHwMonFruType.75 = INTEGER: unknown(1)
SUN-HW-MONITORING-MIB::sunHwMonParentFruIndex.75 = INTEGER: -1
SUN-HW-MONITORING-MIB::sunHwMonParentFruName.75 = STRING:
```

This is expected behavior because this information is specific to Oracle ILOM 3.0. In this case, the -1 signifies nodef.

sunHwMonSensorGroup

In the following example, the numeric sensor MB/V_+12V is retrieved.

▼ How to Retrieve the Sensor Group Information

- **At the command prompt, type the following:**

```
# snmpwalk -v2c -c public -mALL localhost\ SUN-HW-MONITORING-MIB::  
sunHwMonSensorGroup | grep '\.9 = '
```

where grep '\.9 = ' is filtering a property of the FRU we are interested in.

You should see output similar to the following:

```
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorType.9 = INTEGER: voltage  
(133)  
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorName.9 = STRING: /SYS/MB/V_  
+12V  
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorParentFruIndex.9 = INTEGER:  
146  
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorParentFruName.9 = STRING: /  
SYS/MB  
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorAlarmStatus.9 = INTEGER:  
cleared(1)  
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorStateDescr.9 = STRING: Normal  
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorCurrentValue.9 = INTEGER:  
12160  
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorBaseUnit.9 = INTEGER: volts  
(4)  
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorExponent.9 = INTEGER: -3  
SUN-HW-MONITORING-MIB::  
sunHwMonNumericVoltageSensorUpperNonRecoverableThreshold.9 = INTEGER: 14994  
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorUpperCriticalThreshold.9 =  
INTEGER: 13986  
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorUpperNonCriticalThreshold.9 =  
INTEGER: 12978  
SUN-HW-MONITORING-MIB::  
sunHwMonNumericVoltageSensorLowerNonRecoverableThreshold.9 = INTEGER: 8946  
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorLowerCriticalThreshold.9 =  
INTEGER: 9954  
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorLowerNonCriticalThreshold.9 =  
INTEGER: 10962  
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorEnabledThresholds.9  
= BITS: FC lowerThresholdNonCritical(0) upperThresholdNonCritical(1)
```

```
lowerThresholdCritical(2) upperThresholdCritical(3) lowerThresholdFatal(4)
upperThresholdFatal(5)
```

Note - If the server uses Oracle ILOM 2.0, the following lines are returned:

```
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorType.9 = INTEGER: unknown(1)
```

```
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorParentFruIndex.9 = INTEGER:
-1
```

```
SUN-HW-MONITORING-MIB::sunHwMonNumericVoltageSensorParentFruName.9 = STRING:
```

This is expected behavior because this information is specific to Oracle ILOM 3.0.

Tip - When analyzing the following lines, do not forget that the `sunHwMonNumericVoltageSensorCurrentValue` is returned using the exponent set in `sunHwMonNumericVoltageSensorExponent`.

```
SUN-HW-MONITORING-MIB::
```

```
sunHwMonNumericVoltageSensorCurrentValue.9 = INTEGER: 12290
```

```
SUN-HW-MONITORING-MIB::
```

```
sunHwMonNumericVoltageSensorBaseUnit.9 = INTEGER: volts(4)
```

```
SUN-HW-MONITORING-MIB::
```

```
sunHwMonNumericVoltageSensorExponent.9 = INTEGER: -3
```

This example has an exponent of -3, which means that the voltage value of `sunHwMonNumericVoltageSensorCurrentValue` has to be multiplied by 10^{-3} , resulting in 12.290 volts.

sunHwMonIndicatorLocator

You can get and set the `sunHwMonIndicatorLocator`. The following example sets the `sunHwMonIndicatorLocator` to integer(i) value 7, which means `fastBlink` for this OID (Object Identifier).

▼ How to Set the Indicator Locator

- **At the command prompt, type the following:**

```
# snmpset -v2c -c public -mALL localhost\ SUN-HW-MONITORING-MIB::
sunHwMonIndicatorLocatorCurrentStatus.0 i 7
```

You should see output similar to the following:

```
SUN-HW-MONITORING-MIB::sunHwMonIndicatorLocatorCurrentStatus.0 = INTEGER:
fastBlinking(7)
```

Generating SNMP Traps

The combination of Hardware Management Agent and Hardware SNMP Plugins enables you to generate SNMP traps. To test this, you can use IPMItool, which is a component of Hardware Management Pack, to inject a simulated fault. This causes the Hardware SNMP Plugins to generate an SNMP fault.

▼ How to Inject a Simulated Fault



Caution - This procedure returns test SNMP traps, however the values received might not match the values you expect to see when a real SNMP trap is generated. This does not impact non-test SNMP trap functionality.

1. **At the command prompt, type:**

```
ipmitool -U user -P password -H hostname -v sdr list
```

Choose a sensor from the returned list that you want to inject a simulated fault to. In this example the IPMI event: 'P0/VTT' unc assert is used.

2. **At the command prompt, type:**

```
# ipmitool -U user -P password -H hostname event 'P0/VTT' unc assert
```

This injects the IPMI event: 'P0/VTT' unc assert.

You should receive an SNMP trap similar to the following:

```
sysUpTime.0 = Timeticks: (4300) 0:00:43.00
snmpModules.1.1.4.1.1 = OID: sunHwTrapVoltageNonCritThresholdExceeded
sunHwTrapSystemIdentifier.0 = STRING: sg-prg-x6220-01-sp0
```

```
sunHwTrapChassisId.0 = STRING: 1005LCB-0728YM01R7::0739AL71EA
sunHwTrapProductName.0 = STRING: SUN BLADE 6000 MODULAR SYSTEM::SUN BLADE X6220
SERVER MODULE
sunHwTrapComponentName.0 = STRING: /SYS/MB/P0/VTT
sunHwTrapThresholdType.0 = INTEGER: upper(1)
sunHwTrapThresholdValue.0 = STRING:
sunHwTrapSensorValue.0 = STRING:
sunHwTrapAdditionalInfo.0 = STRING: Upper Non-critical going high
sunHwTrapAssocObjectId.0 = OID: zeroDotZero
sunHwTrapSeverity.0 = INTEGER: nonCritical(4)
```

You can verify the SNMP trap by checking the syslog record, which should contain something similar to the following:

```
sg-prg-x6250-01 hwagentd[3470]: P0/VTT (Sensor ID: 0x1b) (Record ID: 0x821):
Upper Non-critical going high.
```

The messages stored in syslog correspond exactly to the SNMP traps. The messages are logged with facility daemon and level notice.

Note - If records corresponding to SNMP traps are not being stored on Oracle Solaris operating systems, make sure that the daemon facility and notice level are enabled.

Using the `itpconfig` Tool

The `itpconfig` tool enables you to configure a trap proxy to send SNMP traps from Oracle Integrated Lights Out Manager (ILOM) over the Host-to-ILOM Interconnect and forward the traps from the host server to a configurable destination. `itpconfig` can also enable or disable the Host-to-ILOM Interconnect, which is available on the latest Oracle servers. The Host-to-ILOM Interconnect provides a high speed internal interconnection between your server's Oracle ILOM service processors and the host, and must be enabled for the trap forwarding to function.

Using `itpconfig` is similar in usage to the Oracle Server CLI Tools. See [“Oracle Server CLI Tools for Oracle Solaris 11.2 User’s Guide”](#) for more information.

- [“itpconfig Command Usage” on page 41](#)
- [“itpconfig Usage Scenario” on page 42](#)
- [“Host-to-ILOM Interconnect Configuration Commands” on page 43](#)
- [“itpconfig Trap Forwarding Commands” on page 44](#)

`itpconfig` Command Usage

The `itpconfig` commands must be run in administrator mode. The default installed location for `itpconfig` is in:

```
/usr/sbin/
```

When a command fails, it returns one of several failure codes listed in [“Error Codes” on page 42](#).

Options

The following options are available to all CLI Tools commands including `itpconfig`:

Short Option	Long Option	Description
-h	--help	Displays help information.

Short Option	Long Option	Description
-V	--version	Displays the tool version.
-q	--quiet	Suppresses informational message output and returns only error codes.

Subcommands

The available itpconfig subcommands are:

Subcommand	Description
list	Show Oracle ILOM trap proxy or Host-to-ILOM Interconnect settings.
modify	Modify Oracle ILOM trap proxy settings.
enable	Enable trap forwarding or Host-to-ILOM Interconnect.
disable	Disable trap forwarding or Host-to-ILOM Interconnect.

See also [“CLI Tools Command Syntax and Conventions”](#) in [“Oracle Server CLI Tools for Oracle Solaris 11.2 User’s Guide ”](#).

Error Codes

itpconfig generates error codes in a similar way to the Oracle Server CLI Tools. See [“CLI Tools Error Codes”](#) in [“Oracle Server CLI Tools for Oracle Solaris 11.2 User’s Guide ”](#).

In addition, itpconfig generates the following error codes:

Code Number	Error Description
81	Oracle ILOM SNMP timeout.
82	Oracle ILOM SNMP failure.

These errors can occur if there are issues communicating with the Oracle ILOM SNMP service when enabling the trap proxy.

itpconfig Usage Scenario

The high level steps for enabling fault forwarding are:

1. Install the Oracle Hardware Management Agent and SNMP Plugins packages.
See [“Oracle Hardware Management Pack for Oracle Solaris 11.2 Installation Guide ”](#)
These packages contain all the necessary software for `itpconfig`.
2. The Host-to-ILOM Interconnect is required for `itpconfig` to function (and is enabled by default for Oracle Solaris operating system).
If the Host-to-ILOM Interconnect is not enabled for your system, you can use `itpconfig` to enable it, see [“How to Enable Host-to-ILOM Interconnect” on page 43](#).
3. Enable the ILOM trap proxy.
See [“How to Enable Trap Forwarding” on page 45](#)

Note - `itpconfig` uses ILOM Notification Alert Rule 15 to set up the trap forwarding. If this alert rule is in use, `itpconfig` fails. See [“itpconfig Troubleshooting” on page 49](#) for a work around.

4. Start or restart the SNMP service daemon on the server.
Refer to your OS documentation.
5. Ensure that SNMP has been configured in Oracle ILOM so that traps can be generated.
Refer to your Oracle ILOM documentation (<http://www.oracle.com/goto/ilom/docs>).
6. Start a trap listener on the destination server configured to listen to traps from the port and community described in the `itpconfig` arguments.
Any faults generated by the service processor should now generate an SNMP trap which are sent to the destination SNMP trap listener.

Host-to-ILOM Interconnect Configuration Commands

The following procedures are covered in this section:

- [“How to Enable Host-to-ILOM Interconnect” on page 43](#)
- [“How to Disable Host-to-ILOM Interconnect” on page 44](#)
- [“How to List the Host-to-ILOM Interconnect Settings” on page 44](#)

▼ How to Enable Host-to-ILOM Interconnect

The Host-to-ILOM Interconnect is automatically enabled and configured in the Oracle Solaris operating system.

Alternatively, you can use `itpconfig` to manage this feature and its properties.

Note - It is recommended that you use this command without any arguments and let `itpconfig` choose the settings. You can override the defaults with different IP and netmask addresses, but this is for advanced users only.

- **Issue the following command:**

```
itpconfig enable interconnect [--ipaddress=ipaddress] [--netmask=netmask] [--hostipaddress=hostipaddress]
```

Option	Description	Example
--ipaddress	Oracle ILOM IP address. This address must be in the format: 169.254.x.x	169.254.175.72
--netmask	Oracle ILOM netmask.	255.255.255.0
--hostipaddress	Host IP address. This address must be in the format: 169.254.x.x	169.254.175.73

▼ How to Disable Host-to-ILOM Interconnect

To disable the Host-to-ILOM Interconnect between the host and Oracle ILOM, use the `itpconfig disable interconnect` command.

- **Issue the following command:**

```
itpconfig disable interconnect
```

▼ How to List the Host-to-ILOM Interconnect Settings

To list the Host-to-ILOM Interconnect state and IP settings on both the Oracle ILOM and host side of the interconnect, use `itpconfig list interconnect`.

- **Issue the following command:**

```
itpconfig list interconnect
```

itpconfig Trap Forwarding Commands

This section includes the following procedures:

- [“How to Enable Trap Forwarding” on page 45](#)
- [“How to Disable Trap Forwarding” on page 45](#)
- [“How to Disable Trap Forwarding” on page 45](#)

▼ How to Enable Trap Forwarding

- To enable trap forwarding, issue the following command:

```
itpconfig enable trapforwarding --ipaddress=ipaddress --port=port --community=community
```

Note - If the trap forwarding is already enabled, use the `itpconfig modify trapforwarding` command instead.

Mandatory options for `itpconfig enable trapforwarding` are:

Option	Description
<code>--ipaddress</code>	Sets the destination IP address for the forwarded trap. This can be loopback (127.0.0.1) or any other valid IP address. This must correspond to the configuration of the SNMP listener.
<code>--port</code>	Sets the destination port for the forwarded trap. There is no default value, but 162 is a common port value. This must correspond to the configuration of the SNMP listener.
<code>--community</code>	Sets the destination SNMP V2c community for the forwarded trap. This value must correspond to the configuration of the SNMP listener.

Example:

```
itpconfig enable trapforwarding --ipaddress=127.0.0.1 --port=1234 --community=test
```

▼ How to Disable Trap Forwarding

- To disable `itpconfig trap forwarding`, issue the following command:

```
itpconfig disable trapforwarding
```

The `disable` command takes no additional parameters and disables the trap forwarding operation on both ILOM and the host.

▼ How to List Trap Forwarding Settings

- To list `itpconfig trap forwarding` settings, issue the following command:

itpconfig list trapforwarding

This returns output similar to the following:

```
Trap Forwarding
=====
Trap Forwarding is enabled
Trap Forwarding Destination: 127.0.0.1
Trap Forwarding Port: 162
Trap Forwarding Community: test
```

The list command takes no additional parameters.

Using Oracle Hardware Management Pack to Monitor Disk Diagnostic Events

This section describes enhanced diagnostic features added to Oracle Hardware Management Pack to collect disk error and SMART events from disks attached to the Sun Storage 6 Gb SAS PCIe HBA, Internal (SGX-SAS6-INT-Z) and store them in the hardware management agent event log.

- [“Monitoring Disk Events” on page 47](#)

Monitoring Disk Events

As of Oracle Solaris 11.2 with SRU 10, Oracle Hardware Management Pack now includes enhanced diagnostic features to collect disk error and SMART events from disks attached to the Sun Storage 6 Gb SAS PCIe HBA, Internal (SGX-SAS6-INT-Z), whether independent or in a RAID volume.

These enhanced diagnostic events are captured and logged in `/var/log/ssm/event.log` when the hardware management agent is running.

The following table lists the enhanced diagnostic events being logged.

Event Name in Log	Description
PD_RECOVERED_ERROR	A disk recovered error was detected.
PD_BAD_DEVICE_FAULT	A non-recoverable drive failure was detected by the device while performing a command.
PD_MEDIA_ERROR	A medium error was detected by the device that was non-recoverable.
PD_DEVICE_ERROR	A non-recoverable hardware failure was detected by the device. The device may be offlined or degraded.
PD_TRANSPORT_ERROR	A path to the device has been unconfigured due to transport instability.
PD_OVER_TEMPERATURE	Disk SMART process reports a critical temperature.
PD_SELF_TEST_FAILURE	One or more disk SMART self tests failed.

Event Name in Log	Description
PD_PREDICTIVE_FAILURE	SMART health-monitoring firmware reported that a disk failure is imminent.

The controller polls each physical disk at regular intervals. If a disk has encountered an error, an event is generated by the controller. The hardware management agent captures that event and enters it in the hardware management event log.

To view the event information in the hardware management event log, type:

```
# view /var/log/ssm/event.log
```

For enhance diagnostic disk events, you will see information similar to:

```
Thu Apr 30 12:32:31 2015:(CLI) Event Name : PD_MEDIA_ERROR
Thu Apr 30 12:32:31 2015:(CLI) Event Description : A medium error was
detected by the device that was non-recoverable.
Thu Apr 30 12:32:31 2015:(CLI) ASC : 0x10
Thu Apr 30 12:32:31 2015:(CLI) ASCQ : 0x3
Thu Apr 30 12:32:31 2015:(CLI) Sense Key : 0x3
Thu Apr 30 12:32:31 2015:(CLI) Source : LSI
Thu Apr 30 12:32:31 2015:(CLI) SAS Address : 0x5000cca01200fadd
Thu Apr 30 12:32:31 2015:(CLI) LSI Description : Unexpected sense: PD
0c(e0xfc/s1) Path 5000cca01200fadd, CDB: 2f 00 00 fc 4d 42 00 10 00 00,
Sense: 3/10/03
Thu Apr 30 12:32:31 2015:(CLI) Event TimeStamp : 04/30/2015 ; 19:30:25
Thu Apr 30 12:32:31 2015:(CLI) Node ID : 00000000:12
Thu Apr 30 12:32:31 2015:(CLI) Nac Name : /SYS/HDD1
Thu Apr 30 12:32:31 2015:(CLI) Serial Number : 001015N0JPXA PMG0JPXA
Thu Apr 30 12:32:31 2015:(CLI) WWN No : PDS:5000cca01200fadd
Thu Apr 30 12:32:31 2015:(CLI) Disk Model : H106030SDSUN300G
```

You can then use the information in the event listing to determine which physical disk in the system has the issue. Information such as the Oracle ILOM Nac Name (which matches the label on the front panel of the system) and drive Serial Number help you identify the disk and its drive slot in the system.

Note - For PD_OVER_TEMPERATURE, PD_SELF_TEST_FAILURE and PD_PREDICTIVE_FAILURE events, use Oracle ILOM to configure proactive alerts.

For the other disk diagnostic events described in this document, it is up to the administrator to check the hardware management event log for these disk events when a disk problem is suspected. There is currently no alert mechanism to proactively announce these events.

Troubleshooting the Hardware Management Agent

This section provides tips and solutions for the most common problems you might encounter when working with the Hardware Management Agent.

- [“itpconfig Troubleshooting” on page 49](#)
- [“The Hardware Management Agent Continually Goes Into Maintenance State” on page 49](#)
- [“How Can I Determine if SNMP Is Configured Correctly for the Hardware Management Agent?” on page 50](#)
- [“SNMP Data Is Not Available Immediately After Starting the Hardware Management Agent” on page 50](#)
- [“Where Are the MIB Files Located?” on page 51](#)

itpconfig Troubleshooting

itpconfig uses ILOM Notification Alert Rule 15 to set up the trap forwarding. If this alert rule is in use, itpconfig fails with error code 83. This error is caused when you try to run itpconfig when ILOM Notification Alert Rule 15 is already defined on the system.

To work around this, set the destination IP address of ILOM Notification Alert Rule 15 to 0.0.0.0.

The Hardware Management Agent Continually Goes Into Maintenance State

Logs show that `svc:/system/sp/management:default` continually restarts (see `/var/log/ssm/hwmgmtd.log`).

This can be due to an inability to establish an IPMI connection to the service processor.

To check IPMI connection, run the following command:

```
# /usr/sbin/ipmitool lan print 1
```

If there is no output, then the connection is down.

Power off the system, including the service processor (this could require unplugging the system). Then power on and restart the system. This should get hardware management agent working, but the problem will likely reappear.

Upgrade to the latest ILOM/Firmware package for the system. The issue is a known problem with the SPARC system firmware and its interaction with some versions of Oracle ILOM. This issue is fixed in all current firmware releases for all platforms.

How Can I Determine if SNMP Is Configured Correctly for the Hardware Management Agent?

The `hmp-snmp` package must be installed in order to use the SNMP functionality with the hardware management agent (`svc:/system/sp/management:default`). To check to see if the `hmp-snmp` package is installed, type the command:

```
pkg list | grep system/management/hmp-snmp
```

Packages with an "i" under the IFO column indicate that the package has been installed.

SNMP configuration needs to be done manually. See [“Configuring Hardware Management Agent and Hardware SNMP Plugins”](#).

SNMP Data Is Not Available Immediately After Starting the Hardware Management Agent

The hardware management agent service, `svc:/system/sp/management:default`, runs on a 5 minute polling cycle. Since Initialization of the hardware management agent can take up to 5 minutes, depending on the system configuration, no data will be available through SNMP queries during that time.

Wait a full 5 minutes after a system boot or agent restart before attempting to view any SNMP data.

Where Are the MIB Files Located?

The MIB files available with Oracle Hardware Management Pack are installed in:

`/usr/lib/asm/lib/mibs`

Index

C

- command usage
 - itpconfig, 41
- Configuration File
 - Hardware Management Agent, 17
- Configure
 - Hardware Management Agent, 17
 - Host Operating System's SNMP, 19
 - Log Level, 17
 - SNMP Gets, 20
 - SNMP Sets, 20
 - SNMP Traps, 20
- Configure Net-SNMP
 - Solaris, 19

D

- disk events, 47
- documentation links, 9

F

- feedback, 9

H

- Hardware Management Agent
 - Configuration File, 17
 - Configure, 17
 - Configure SNMP, 19
 - Log File, 17
- Hardware SNMP Plugins, 23
- Host-to-ILOM Interconnect
 - disabling, 44
 - enabling, 43

- listing, 44
- hwagentd_log_levels
 - Parameter, 17
- hwmgmtd.conf, 17
- hwmgmtd.log, 17

I

- ILOM Notification Alert Rule 15, 49
- ILOM Trap Proxy
 - overview, 16
- IPMItool, 38
- itpconfig
 - command usage, 41
 - overview, 16
- itpconfig troubleshooting, 49

L

- Linux
 - SNMP Gets, 20
 - SNMP Sets, 20
- local Oracle ILOM interconnect *See* Host-to-ILOM Interconnect
- Log File
 - Hardware Management Agent, 17
- Log Level
 - Configure, 17

M

- Management Information Base, 23
 - Sun Hw Monitoring, 23
 - Sun Hw Trap MIB, 27
- MIB *See* Management Information Base

monitoring disk events, 47

N

net-snmp
 restarting after configuration, 21

O

Oracle Server Hardware Management Agent
 overview, 15
Oracle Server Hardware SNMP Plugins, 16
 overview, 15
Oracle Server Management Agent
 overview, 15
Oracle Solaris OS
 restarting Net-SNMP, 21
Overview
 Oracle Server Hardware Management Agent, 15
 Oracle Server Hardware SNMP Plugins, 15

S

Sensor
 Severity, 25
Severity
 Sensor, 25
SNMP, 15
 Configure, 19
 Generating Traps, 38
 Retrieving and Setting Information Through, 31
SNMP Gets, 20
SNMP Sets, 20
SNMP Traps, 20
snmpd.conf, 19, 20, 20, 20
snmpwalk, 31
Solaris
 Configure Net-SNMP, 19
 SNMP Sets, 20
 SNMP Traps, 20
Sun Hw Monitoring MIB
 Overview, 23
Sun Hw Trap MIB
 Overview, 27
sunHwMonMIB

 overview, 16
sunHwTrapMIB
 overview, 16
sunStorageMIB
 overview, 16
Syslog, 38
System Event Log, 15

T

Troubleshooting, 47, 49