Oracle Server Management Agent for Oracle Solaris 11.4 User's Guide

E79573-03
May 2023

Oracle Server Management Agent for Oracle Solaris 11.4 User's Guide,

E79573-03

Primary Author: Ralph Woodley

Contents

Using This Documentation

Product Documentation Library vii
Feedback vii
Change History vii

1 Oracle Hardware Management Agent Overview

2 Hardware Monitoring Using the Server Management Agent

Oracle Server Hardware Management Agent 2-1
Oracle Server Hardware SNMP Plugins 2-2
itpconfig and the Oracle ILOM Trap Proxy 2-3

3 Configuring the Hardware Management Agent and Hardware SNMP Plugins

Hardware Management Agent Configuration File 3-1
- Management Agent Resource Monitoring, Polling and SNMP Support 3-2
- Log Levels: Basic and Bit Flag Parameters 3-4

Configure the Hardware Management Agent for SNMP Support and Logging Level 3-6
Configuring Net-SNMP In Oracle Solaris 3-6
- How to Configure SNMP Gets 3-7
- How to Configure SNMP Sets 3-7
- How to Configure SNMP Traps 3-7
- How to Restart Net-SNMP 3-8

4 Oracle Server Hardware SNMP Plugins Overview

Overview of Sun HW Monitoring MIB 4-1
- Sun Server Product and Chassis 4-2
- Sun Server Service Processor 4-2
- Sun Server Hardware Monitoring MIB 4-2

Sun Server Hardware Management Agent 4-2
Sun Server Hardware Inventory 4-3
Sun Server Hardware Monitor Sensor Group 4-3
sunHwMonIndicatorGroup 4-4
sunHwMonTotalPowerConsumption 4-5
Overview of Sun HW Trap MIB 4-5
Overview of Sun Storage MIB 4-5
Sun Storage MIB Objects 4-6
Physical and Logical Storage Objects 4-6

5 Working With the Hardware Management Agent

Retrieving and Setting Information Through SNMP 5-1
sunHwMonProductGroup 5-2
How to Retrieve the Product Information from an Oracle x86 Server 5-2
How to Retrieve The Product Information on an Oracle x86 Server Module 5-3
sunHwMonProductChassisGroup 5-3
How to Retrieve the Server Module's Product Chassis Information 5-3
sunHwMonSPGroup 5-4
How to Retrieve Service Processor Information 5-4
sunHwMonInventoryTable 5-5
How to Retrieve Inventory Information 5-5
sunHwMonSensorGroup 5-6
How to Retrieve the Sensor Group Information 5-6
sunHwMonIndicatorLocator 5-8
How to Set the Indicator Locator 5-8
Generating SNMP Traps 5-8
How to Inject a Simulated Fault 5-9

6 Configuring Oracle ILOM SNMP Trap Forwarding Using itpconfig

itpconfig Command Usage 6-1
Subcommands 6-1
Supported Types 6-2
Options 6-2
Error Codes 6-2
itpconfig Usage Scenario 6-3
Host-to-ILOM Interconnect Configuration Commands 6-3
How to Enable Host-to-ILOM Interconnect 6-4
How to Disable Host-to-ILOM Interconnect 6-4
How to List the Host-to-ILOM Interconnect Settings 6-4

itpconfig Trap Forwarding Commands 6-4
How to Enable Trap Forwarding 6-5
How to Disable Trap Forwarding 6-5
How to List Trap Forwarding Settings 6-5

7 Using Oracle Hardware Management Pack to Monitor Disk Diagnostic Events

Monitoring Disk Events 7-1

8 Troubleshooting the Hardware Management Agent

General Management Agent Troubleshooting 8-1
itpconfig Troubleshooting 8-1
The Hardware Management Agent Continually Goes Into Maintenance State 8-1
How Can I Determine if SNMP Is Configured Correctly for the Hardware Management Agent? 8-2
SNMP Data Is Not Available Immediately After Starting the Hardware Management Agent 8-2
Where Are the MIB Files Located? 8-2

Index

Using This Documentation

- **Overview** – Describes how to install the software
- **Audience** – Technicians, system administrators, and authorized service providers
- **Required knowledge** – Advanced experience troubleshooting and replacing hardware
- Product Documentation Library
- Feedback
- Change History

Product Documentation Library

Documentation and resources for this product and related products are available at https://www.oracle.com/goto/ohmp/solarisdocs.

Feedback

Provide feedback about this documentation at https://www.oracle.com/goto/docfeedback.

Change History

The following changes have been made to the document.

- August 2018. Initial publication.
- March 2019. Updated *Management Agent User's Guide* to describe new configurable management agent resource monitoring threads added to `hwmgmtd.conf` file.
- May 2023. Minor documentation formating edits.

1

Oracle Hardware Management Agent Overview

Oracle Hardware Management Pack includes a rich set of command line interface tools and agents that are run from your host operating system to configure and monitor server hardware. For information on operating system and server support for each Oracle Hardware Management Pack component, refer to the support matrix available at https://www.oracle.com/goto/ohmp.

Oracle Hardware Management Pack for Oracle Solaris is an integrated component of the Oracle Solaris 11.4 operating system. Do not download and use other versions of Oracle Hardware Management Pack that are not specifically qualified for the Oracle Solaris 11.4. For information on installing the Hardware Management Agent, see .

If you have Oracle Solaris 11.1 or earlier or other operating systems, continue to use Oracle Hardware Management Pack, available as a separate download from https://support.oracle.com.

> **Note:**
>
> This documentation applies to servers running the Oracle Solaris 11.4 operating system.

This guide includes information on the Oracle Hardware Management Agent:

Description	Link
Introduction to the Oracle Hardware Management Agent	Hardware Monitoring Using the Server Management Agent
Configure the Hardware Management Agent with SNMP.	Configuring the Hardware Management Agent and Hardware SNMP Plugins
Learn about the available SNMP plugin MIBs.	Oracle Server Hardware SNMP Plugins Overview
Use the Hardware Management Agent with SNMP plugins to obtain information about your system.	Working With the Hardware Management Agent
Setup the Oracle ILOM trap proxy.	Configuring Oracle ILOM SNMP Trap Forwarding Using itpconfig
Learn about disk events and logs for select SAS HBAs.	Using Oracle Hardware Management Pack to Monitor Disk Diagnostic Events
How to troubleshoot issues with the Hardware Management Agent.	Troubleshooting the Hardware Management Agent

2

Hardware Monitoring Using the Server Management Agent

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

The Oracle Server Hardware Management Agent components include:

- Oracle Server Hardware Management Agent – Runs in the background to collect information about the system to support Oracle ILOM and SNMP monitoring.
- Oracle Server SNMP Plugins – System Management Information Base (MIBs) that support native SNMP monitoring.
- Oracle ILOM SNMP trap forwarding – The `itpconfig` utility allows you to set up a proxy on the host to forward SNMP traps received in Oracle ILOM to an SNMP trap destination of your choice over the Host-to-ILOM Interconnect.

> **Note:**
>
> The SNMP monitoring function of the agent is disabled by default and must be enabled and configured by the user as described in this document.

For additional information about these components, see:

- Oracle Server Hardware Management Agent
- Oracle Server Hardware SNMP Plugins
- itpconfig and the Oracle ILOM Trap Proxy

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 and server module's hardware using an operating system native SNMP 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.

> **Note:**
>
> By default, SNMP functionality for the agent is disabled even if you installed the SNMP Plugins. This functionality can be enabled through the `hwmgmtd.conf` file. For more information, see Configuring the Hardware Management Agent and Hardware SNMP Plugins.

In addition, the Hardware Management Agent 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, these include versions of hardware-specific Management Information Bases (MIB) which have been designed to enable you to monitor your Oracle servers effectively.

> **Note:**
>
> Even with the appropriate Oracle Hardware Management Pack for Solaris components installed, you still need to configure the Hardware Management Agent to support SNMP requests. See Configuring the Hardware Management Agent and Hardware SNMP Plugins.

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 Oracle ILOM Trap Proxy

The `itpconfig` command line tool allows you to configure a trap forwarding proxy to forward Oracle ILOM generated SNMP traps to the host or a configured SNMP trap destination over the Host-to-ILOM Interconnect. This allows SNMP traps to be forwarded to a destination you specify without having to have a network connection to the server's NET MGT port.

> **Note:**
>
> The Hardware Management Agent supports a trap-triggered polling feature that will only work with host-based SNMP monitoring in conjunction with the Oracle LOM trap proxy. For more information about this feature, see Hardware Management Agent Configuration File.

The `itpconfig` command can be used to both set up a trap proxy and to configure the Host-to-ILOM Interconnect between the Oracle ILOM service processor and the host.

Refer to your server documentation to see if your server supports the Host-to-ILOM Interconnect.

For information on setting up the Oracle ILOM trap proxy, see Configuring Oracle ILOM SNMP Trap Forwarding Using itpconfig.

3

Configuring the Hardware Management Agent and Hardware SNMP Plugins

This section provides instructions for configuring the Hardware Management Agent and Hardware SNMP Plugins, as well as information about using Hardware Management Agent successfully. The section contains the following:

- Hardware Management Agent Configuration File
- Configure the Hardware Management Agent for SNMP Support and Logging Level
- Configuring Net-SNMP In Oracle Solaris

Hardware Management Agent Configuration File

By default, the Hardware Management Agent manages the storage poller and provides Oracle ILOM information about storage devices in the system including any changes to these devices.

The trap proxy notifies the agent when Oracle ILOM has sent a trap. Instead of regularly polling Oracle ILOM for information, detection of a trap will initiate an update cycle to get the newest data from Oracle ILOM. By default, if no traps trigger a polling cycle then Oracle ILOM is polled by the Hardware Management Agent once per hour.

> **Note:**
>
> The trap-based polling feature will only work if the host is configured for SNMP monitoring and the Oracle LOM trap proxy has been set up. For information on setting up the Oracle ILOM trap proxy, see Configuring Oracle ILOM SNMP Trap Forwarding Using itpconfig.

If more precise real-time data of sensor values (temperature/voltage) is required, a custom "polling_round_delay" configuration option can be added to the `/etc/ssm/hwmgmtd.conf` file. Adding this option into the configuration file allows you to change the default polling interval.

For SNMP monitoring, the Hardware Management Agent can be configured as an SNMP agent for the SUN-HW-MONITORING-MIB and SUN-STORAGE-MIB that are part of Oracle Hardware Management Pack. This enables the native SNMP service running on the host to query information stored in these MIBs. By default, this feature is turned off but can be configured in the `hwmgmtd.conf` configuration file.

To prevent possible memory leaks that can cause performance issues, the Hardware Management Agent service (`svc:/system/sp/management:default`) can be configured to monitor its own memory usage and reset itself if a configurable threshold is crossed.

The Hardware Management Agent also records log messages in the `/var/log/ssm/hwmgmtd.log` file. These messages can be used to troubleshoot the running status of the Hardware Management Agent. Log level configuration is done using *basic* or *bit flag* parameters.

The following subsections describe the configurable parameters in the `hwmgmtd.conf` file. For information on performing the configuration, see Configure the Hardware Management Agent for SNMP Support and Logging Level.

- Management Agent Resource Monitoring, Polling and SNMP Support
- Log Levels: Basic and Bit Flag Parameters

Management Agent Resource Monitoring, Polling and SNMP Support

The following table lists the configurable values for polling and thread selection in the `hwmgmtd.conf` file when SNMP monitoring is used on the host. In addition, there is a memory usage monitoring setting for the management agent to prevent possible memory leak issues.

> **Note:**
>
> The default configuration for polling provides the optimal functionality in systems where SNMP monitoring is not used. Where host-based SNMP monitoring is configured and the Oracle ILOM trap proxy setup using `itpconfig`, the Hardware Management Agent will perform an initial poll of Oracle ILOM and then listen for Oracle ILOM SNMP traps to decide when a re-pole is needed. Enabling SNMP polling threads increase resources used by the Hardware Management Agent which might affect system performance.

Thread	Description
`storage_poller_enabled=<value>`	When enabled, the Hardware Management Agent provides up-to-date information about host internal storage to Oracle ILOM. In addition, if SNMP monitoring is enabled on the host, this poller responds to host SNMP requests to the SUN-STORAGE-MIB. Supported values are: 0=disabled; 1=enabled. The default is 1. **Note:** Disabling this parameter is not recommended as it will prevent storage information from being sent to Oracle ILOM.
`inventory_poller_enabled=<value>`	When enabled, this poller responds to host SNMP requests to the SUN-HW-MONITORING-MIB. Supported values are: 0=disabled; 1=enabled. The default is 0.
`request_thread_enabled=<value>`	When enabled, the Hardware Management Agent responds to host SNMP requests and listens for SNMP traps generated by Oracle ILOM. Supported values are: 0=disabled; 1=enabled. The default is 0.
`polling_round_delay=<time in seconds>`	If needed, this option can be added. When set, this option overrides the default polling delay of every hour, or sooner if a trap is detected when using host-based SNMP monitoring and the Oracle ILOM trap proxy. You might want to add this option if you require more precise real-time data regarding sensor values (temperature/voltage). Set the default sleep time in seconds. **Note:** When using this option, Oracle recommends not setting it to anything under 60 seconds due to the additional use of system resources.

Thread	Description
`hwmgmtd_reset_hour=<hour>`	To prevent possible memory leaks, the management agent service should be reset once daily. Choose a time when there is little server activity since restarting the agent temporarily increases I/O activity as it performs device discovery. Set the reset time in whole hours. Supported values are: 0-23. The default is 3 (for 3AM local time). **Note:** In Oracle Solaris 11.4 release SRU 6, this feature has been removed. To monitor possible memory leaks, use the resource_monitor_enabled and resource_monitor_threshold threads.
`resource_monitor_enabled=<value>`	When enabled, the Hardware Management Agent checks for possible memory leaks caused by unusual memory usage by the agent service itself. Supported values are: 0=disabled; 1=enabled. The default is 1. *This thread was introduced in Oracle Solaris 11.4 SRU 6.*
`resource_monitor_threshold=<value>`	When the agent resource monitor is enabled, this setting becomes the basis of an algorithm to define a threshold where the agent will automatically restart if the threshold is reached. The value you enter is used to determine how much memory used by the agent service is too much. For example, if the value you enter is 3, then the threshold would be three times the amount of memory typically required for the agent at service start-up. Set the threshold value in whole numbers. The default is 5 (for five times the typical memory usage at service start-up). *This thread was introduced in Oracle Solaris 11.4 SRU 6.*

Log Levels: Basic and Bit Flag Parameters

The following table lists the supported *basic* values for the `-hwagentd_log_levels` section in the `hwmgmtd.conf` file.

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

Using *bit flag* parameters allows you to set the logging level with a finer level of granularity. The following table lists supported values.

> **Note:**
>
> Oracle recommends that you use the basic logging levels. The bit flag parameters 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 0x003f 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.

Configure the Hardware Management Agent for SNMP Support and Logging Level

1. Find the hwmgmtd.conf file and open it for editing.

   The file is located under:

   `/etc/ssm/hwmgmtd.conf`

2. Find the hwagentd_log_levels parameter and change the logging level to one of the options described in the Hardware Management Agent Configuration File.
3. Find the polling and thread parameters and enable them as described in the Hardware Management Agent Configuration File.
4. Save the modified hwmgmtd.conf file.
5. Disable and re-enable the Hardware Management Agent, which forces the hwmgmtd.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 `hwmgmtd.conf` file with the modified polling and logging level parameters.

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-snmp/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
- How to Configure SNMP Sets
- How to Configure SNMP Traps
- How to Restart Net-SNMP

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 3-1 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

- After configuring snmpd.conf, restart the SNMP daemon using the following command:

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

4

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/ssm/lib/mibs`

> **Note:**
>
> To perform SNMP monitoring using the Hardware SNMP plugins, you must configure polling support in the Hardware Management Agent `hwmgmtd.conf` file as described in Configuring the Hardware Management Agent and Hardware SNMP Plugins.

This section contains the following:

- Overview of Sun HW Monitoring MIB
- Overview of Sun HW Trap MIB
- Overview of Sun Storage MIB

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
- Sun Server Service Processor
- Sun Server Hardware Monitoring MIB
- Sun Server Hardware Management Agent
- Sun Server Hardware Inventory
- Sun Server Hardware Monitor Sensor Group
- sunHwMonIndicatorGroup
- sunHwMonTotalPowerConsumption

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.

sunHwMonSensorGroup Groups	Description
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 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.

sunHwMonIndicatorGroup Groups	Description
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
- Physical and Logical Storage Objects

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

5

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.

> **Note:**
>
> To perform SNMP monitoring using the Hardware SNMP plugins, you must configure polling support in the Hardware Management Agent `hwmgmtd.conf` file as described in Configuring the Hardware Management Agent and Hardware SNMP Plugins.

This section provides the following:

- Retrieving and Setting Information Through SNMP
- sunHwMonProductGroup
- sunHwMonProductChassisGroup
- sunHwMonSPGroup
- sunHwMonInventoryTable
- sunHwMonSensorGroup
- sunHwMonIndicatorLocator
- Generating SNMP Traps

Retrieving and Setting Information Through SNMP

The following section provides some examples of using Net-SNMP's `snmpwalk` utility to get information and the `snmpset` command to set information on 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 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.

> **Note:**
>
> When working with Oracle Hardware Management Pack SNMP MIB files, the default installation location for Oracle Solaris is `/usr/lib/ssm/lib/mibs`. If the examples listed don't work, you might need to append the path using the `-M` option.

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

How to Retrieve the Product Information from an Oracle x86 Server

- At the command prompt, type the following:

  ```
  # snmpwalk -v 2c -c public -m ALL 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 -v 2c -c public -m ALL 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: 080020FFFFFFFFFFFFFF001B24782F9C

  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

How to Retrieve the Server Module's Product Chassis Information

- At the command prompt, type the following:

  ```
  # snmpwalk -v 2c -c public -m ALL 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

How to Retrieve Service Processor Information

- At the command prompt, type the following:

  ```
  # snmpwalk -v 2c -c public -m ALL 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

How to Retrieve Inventory Information

- At the command prompt, type the following:

  ```
  # snmpwalk -v 2c -c public -m ALL 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

How to Retrieve the Sensor Group Information

- At the command prompt, type the following:

  ```
  # snmpwalk -v 2c -c public -m ALL 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

How to Set the Indicator Locator

- At the command prompt, type the following:

  ```
  # snmpset -v 2c -c public -m ALL 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

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
   ```

   Where *user* is the Oracle ILOM user name with root privileges, *password* it the password for the user, and *hostname* is the host name or IP address of the Oracle ILOM SP.

   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.

6

Configuring Oracle ILOM SNMP Trap Forwarding Using itpconfig

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.

Some functionality, such as configuring the Host-to-ILOM Interconnect, is available using either `itpconfig` or `ilomconfig`. For more information on using `ilomconfig`, see .

This section includes the following topics:

- itpconfig Command Usage
- itpconfig Usage Scenario
- Host-to-ILOM Interconnect Configuration Commands
- itpconfig Trap Forwarding Commands

itpconfig Command Usage

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

```
/usr/sbin/
```

The command syntax for `itpconfig` is:

```
itpconfig <subcommand> <type> [options]
```

When a command fails, it returns one of several failure codes listed in Error Codes.

- Subcommands
- Supported Types
- Options
- Error Codes

Subcommands

The available `itpconfig` subcommands are:

Subcommand	Description
`list`	Show Oracle ILOM trap proxy or Host-to-ILOM Interconnect settings.

Subcommand	Description
`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 11.4 User's Guide*.

Supported Types

Type	Description

`interconnect`	Modify Host-to-ILOM interconnect settings.
	Mandatory options for enable or modify include:
	`--ipaddress=`*`ipaddress`* `ILOM interconnect IP address`
	`--hostipaddress=`*`ipaddress`* `Host interconnect IP address`
	`--netmask=`*`netmask`* `Host-to-ILOM interconnect netmask`

`trapforwarding`	Modify ILOM to send SNMP traps for all faults.
	Mandatory options for enable or modify include:
	`--ipaddress=`*`ipaddress`*`IP Address to forward fault traps to`
	`--port=`*`port`* `Port number to forward fault traps to`
	`--community=`*`community`*`SNMP V2c community to use when forwarding fault traps`

Options

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

Short Option	Long Option	Description
`-?, -h`	`--help`	Displays help information.
`-V`	`--version`	Displays the tool version.
`-q`	`--quiet`	Suppresses informational message output and returns only error codes.
`-y`	`--yes`	Execute command without prompting for confirmation.

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 11.4 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.
83	Service Processor has conflicting configuration.

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 SNMP trap forwarding from Oracle ILOM are:

1. Install the Oracle Hardware Management Agent and SNMP Plugins packages.

   See

   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.

3. Enable the ILOM trap proxy.

   See How to Enable Trap Forwarding

   > **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 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 (https://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
- How to Disable Host-to-ILOM Interconnect

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

How to Enable Host-to-ILOM Interconnect

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

If it is not enabled, 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
- How to Disable Trap Forwarding
- How to List Trap Forwarding Settings

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
`--ipaddress`	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.
`--port`	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.
`--community`	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.

7

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

Monitoring Disk Events

Oracle Hardware Management Pack 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.
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.

8

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.

- General Management Agent Troubleshooting
- itpconfig Troubleshooting
- The Hardware Management Agent Continually Goes Into Maintenance State
- How Can I Determine if SNMP Is Configured Correctly for the Hardware Management Agent?
- SNMP Data Is Not Available Immediately After Starting the Hardware Management Agent
- Where Are the MIB Files Located?

General Management Agent Troubleshooting

The best way to troubleshoot problems with Hardware Management Agent is to review the log files.

The Hardware Management Agent stores log information in the `hwmgmtd.log` file.

For more information on the `hwmgmtd.log` file, see Hardware Management Agent Configuration File.

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 the Hardware Management Agent and Hardware SNMP Plugins.

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

The Initialization of the hardware management agent service, `svc:/system/sp/management:default`, 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/ssm/lib/mibs
```

Index

C

command usage
  `itpconfig`, *6-1*
configuration file
  Hardware Management Agent, *3-1*
configure
  agent polling, *3-1*
  Log Level, *3-1*
Configure
  Hardware Management Agent, *3-1*
  Host Operating System's SNMP, *3-6*
  SNMP Gets, *3-7*
  SNMP Sets, *3-7*
  SNMP Traps, *3-7*
Configure Net-SNMP
  Solaris, *3-6*

D

disk events, *7-1*

H

Hardware Management Agent
  configuration file, *3-1*
  Configure, *3-1*
  Configure SNMP, *3-6*
  daily reset value, *3-1*
  log file, *3-1*
  polling, *3-1*
  SNMP support, *3-1*
Hardware SNMP Plugins, *4-1*
Host-to-ILOM Interconnect
  disabling, *6-4*
  enabling, *6-4*
  listing, *6-4*
hwagentd_log_levels
  parameters, *3-1*
hwmgmtd.conf
  configurable options, *3-1*
hwmgmtd.conf parameters
  inventory_poller_enabled, *3-2*
  polling_round_delay, *3-2*
  request_thread_enabled, *3-2*
hwmgmtd.conf parameters *(continued)*
  resource_monitor_enabled, *3-2*
  resource_monitor_threshold, *3-2*
  storage_poller_enabled, *3-2*
hwmgmtd.log, *3-1*

I

ILOM Notification Alert Rule 15, *8-1*
ILOM Trap Proxy
  overview, *2-3*
IPMItool, *5-8*
itpconfig
  command usage, *6-1*
  overview, *2-3*
itpconfig troubleshooting, *8-1*

L

Linux
  SNMP Gets, *3-7*
  SNMP Sets, *3-7*
local Oracle ILOM interconnect, *6-4*
log file
  Hardware Management Agent, *3-1*
log level
  configuring the agent, *3-1*

M

Management Information Base, *4-1*
  Sun Hw Monitoring, *4-1*
  Sun Hw Trap MIB, *4-5*
MIB, *4-1*
monitoring
  message level logging configuration, *3-1*
  polling configuration, *3-1*
monitoring disk events, *7-1*

N

net-snmp
  restarting after configuration, *3-8*

O

Oracle Server Hardware Management Agent
overview, *2-1*
Oracle Server Hardware SNMP Plugins, *2-2*
overview, *2-1*
Oracle Server Management Agent
overview, *2-1*
Oracle Solaris OS
restarting Net-SNMP, *3-8*
Overview
Oracle Server Hardware Management Agent, *2-1*
Oracle Server Hardware SNMP Plugins, *2-1*

S

Sensor
Severity, *4-3*
Severity
Sensor, *4-3*
SNMP, *2-1*
Configure, *3-6*
Generating Traps, *5-8*
retrieving and setting information, *5-1*
SNMP Gets, *3-7*
SNMP Sets, *3-7*
SNMP support
configuring the agent, *3-1*
SNMP Traps, *3-7*
snmpd.conf, *3-6*, *3-7*
snmpwalk, *5-1*
Solaris
Configure Net-SNMP, *3-6*
SNMP Sets, *3-7*
SNMP Traps, *3-7*
Sun Hw Monitoring MIB
Overview, *4-1*
Sun Hw Trap MIB
Overview, *4-5*
sunHwMonMIB
overview, *2-2*
sunHwTrapMIB
overview, *2-2*
sunStorageMIB
overview, *2-2*
Syslog, *5-8*
System Event Log, *2-1*

T

Troubleshooting, *7-1*, *8-1*