

Oracle Hardware Management Pack 2.4 Server CLI Tools User's Guide



E72072-09
May 2023



Oracle Hardware Management Pack 2.4 Server CLI Tools User's Guide,

E72072-09

Copyright © 2017, 2023, Oracle and/or its affiliates.

Primary Author: Ralph Woodley

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, software documentation, data (as defined in the Federal Acquisition Regulation), 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 embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. 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®, Java, and MySQL are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside 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, Epyc, and the AMD 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.

Copyright © 2017, 2023, Oracle et/ou ses affiliés.

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, accorder de licence, 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, la documentation du logiciel, les données (telles que définies dans la réglementation "Federal Acquisition Regulation") ou la documentation qui l'accompagne sont livrés sous licence au Gouvernement des États-Unis, ou à quiconque qui aurait souscrit la licence de ce logiciel pour le compte du Gouvernement des États-Unis, la notice suivante s'applique :

UTILISATEURS DE FIN DU GOUVERNEMENT É.-U. : programmes Oracle (y compris tout système d'exploitation, logiciel intégré, tout programme intégré, installé ou activé sur le matériel livré et les modifications de tels programmes) et documentation sur l'ordinateur d'Oracle ou autres logiciels Oracle Les données fournies aux utilisateurs finaux du gouvernement des États-Unis ou auxquelles ils ont accès sont des "logiciels informatiques commerciaux", des "documents sur les logiciels informatiques commerciaux" ou des "données relatives aux droits limités" conformément au règlement fédéral sur l'acquisition applicable et aux règlements supplémentaires propres à l'organisme. À ce titre, l'utilisation, la reproduction, la duplication, la publication, l'affichage, la divulgation, la modification, la préparation des œuvres dérivées et/ou l'adaptation des i) programmes Oracle (y compris tout système d'exploitation, logiciel intégré, tout programme intégré, installé, ou activé sur le matériel livré et les modifications de ces programmes), ii) la documentation informatique d'Oracle et/ou iii) d'autres données d'Oracle, sont assujetties aux droits et aux limitations spécifiés dans la licence contenue dans le contrat applicable. Les conditions régissant l'utilisation par le gouvernement des États-Unis des services en nuage d'Oracle sont définies par le contrat applicable à ces services. Aucun autre droit n'est accordé au gouvernement américain.

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 un risque de 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 des applications dangereuses.

Oracle®, Java, et MySQL sont des marques déposées d'Oracle Corporation et/ou de ses affiliés. Tout autre nom mentionné peut être une marque appartenant à un autre propriétaire qu'Oracle.

Intel et Intel Inside 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, Epyc, et le logo AMD sont des marques ou des marques déposées d'Advanced Micro Devices. UNIX est une marque déposée de 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 et excluent toute garantie expresse ou implicite 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.

Contents

Using This Documentation

Product Documentation Library	x
Feedback	x
Change History	x

1 Oracle Server CLI Tools Overview

2 Host-to-ILOM Interconnect

3 CLI Tools Command Syntax and Conventions

CLI Tools Command Syntax	3-1
CLI Tools Device-Naming Convention	3-2

4 Using biosconfig to Update the BIOS

biosconfig Command Overview	4-1
biosconfig Requirements	4-1
biosconfig Device Terminology	4-2
Device Name Examples	4-2
Editing XML Files	4-2
biosconfig Command Syntax	4-3
biosconfig for Oracle Solaris OS	4-4
Viewing biosconfig Command Options and Version Information	4-5
View biosconfig Command Options	4-5
View biosconfig Version Information	4-5
Configuring the Device Boot Order	4-6
Methods for Changing the Boot List	4-7
Set the First Boot Device for the Next Boot	4-7
Make a Persistent Change to Boot Order	4-8
Change Boot Order Based on the PCI Bus, Device, or Function	4-9

Configuring the BIOS CMOS	4-10
Capture the BIOS CMOS Golden Image	4-10
Apply the BIOS CMOS Golden Image	4-11
Configuring Individual CMOS Settings	4-12
Static and Dynamic CMOS Settings	4-13
Configure a Static CMOS Setting	4-13
Configure a Dynamic Setting	4-14
Commands That Produce Unrelated, Innocuous, Extra Output	4-14

5 Using fwupdate to Update Firmware

fwupdate Command Overview	5-1
fwupdate Features	5-1
fwupdate Command Prerequisites	5-2
Secure Boot Setting Can Affect fwupdate Commands on Intel-Based Network and Storage Controllers	5-3
Linux MMIO Access Settings Can Affect fwupdate Commands On Intel Network Controllers	5-3
Downloading Firmware Patches	5-4
fwupdate and Service Processor Access	5-5
Obtaining SSL Certificates for TLS Access	5-5
Command Options for Accessing Oracle ILOM Over a Remote Network Connection	5-7
fwupdate Command Syntax	5-10
Listing Component Firmware Information	5-13
list Subcommand Overview	5-14
List All Component Firmware Information	5-18
List Specific Component Firmware Information	5-21
Updating Component Firmware	5-24
update Subcommand Overview	5-24
Updating Component Firmware With a Metadata File (Automatic Mode)	5-25
Update System Firmware Using Automatic Mode	5-30
Update Device Firmware Using Automatic Mode	5-33
Update a SPARC Fallback Image Using Automatic Mode	5-34
Updating Component Firmware Without a Metadata File (Manual Mode)	5-36
Update System Firmware Using Manual Mode	5-41
Update Device Firmware Using Manual Mode	5-43
Reset a Device After a Firmware Update	5-45
Execution Summary	5-46

6 Using hwmgmtcli to Display Hardware Information

hwmgmtcli Command Syntax	6-1
List Subsystem Information	6-2
View Open Problems	6-3
Export Subsystem Information	6-3

7 Using ilomconfig to Configure Oracle ILOM

ilomconfig Command Overview	7-1
ilomconfig Features	7-1
Restoring and Modifying Oracle ILOM XML Configuration Files	7-2
ilomconfig and Service Processor Access	7-3
Obtaining SSL Certificates for TLS Access	7-3
Command Options for Accessing Oracle ILOM Over a Remote Network Connection	7-5
ilomconfig Command Syntax	7-8
Importing and Exporting XML Configurations	7-11
Export an XML Configuration	7-11
Import an XML Configuration	7-13
Listing System and SP Information	7-14
List System Summary Information	7-15
List Users	7-15
List an SNMP Community	7-16
List IPv4 Network Settings	7-16
List IPv6 Network Settings	7-16
List Service Processor Identification Information	7-16
List DNS Information	7-17
List Clock Information	7-17
Modifying Oracle ILOM Configurations	7-17
Restore Oracle ILOM to Defaults	7-18
Create a User	7-18
Delete a User	7-18
Modify a User Password or Role	7-18
Create an SNMP Community	7-19
Modify IPv4 Network Settings	7-19
Modify IPv6 Network Settings	7-20
Modify Identification Information	7-20
Modify DNS Information	7-21
Modify Clock Information	7-22
Configuring the Host-to-ILOM Interconnect	7-22
Enable the Host-to-ILOM Interconnect	7-23

Disable the Host-to-ILOM Interconnect	7-23
Modify the Host-to-ILOM Interconnect	7-23
List the Host-to-ILOM Interconnect Settings	7-24
Verify the Host-to-ILOM Interconnect Settings	7-24
Managing Oracle ILOM and Host Watchdog	7-24
HMP Watchdog Overview	7-25
HMP Watchdog Agent Overview	7-25
ILOM Watchdog Overview	7-26
Host Watchdog Overview	7-27
ILOM Watchdog ilomconfig Commands	7-28
List Current ILOM Watchdog Settings	7-29
Enable ILOM Watchdog	7-29
Disable ILOM Watchdog	7-29
Modify Current ILOM Watchdog Settings	7-29
Reset ILOM Watchdog Parameters to Factory Defaults	7-30
Host Watchdog ilomconfig Commands	7-30
List Current Host Watchdog Settings	7-30
Enable Host Watchdog	7-31
Disable Host Watchdog	7-31
Modify Current Host Watchdog Settings	7-31
Reset Host Watchdog Parameters to Factory Defaults	7-31
Troubleshooting ILOM Watchdog and Host Watchdog States	7-31
Start and Stop Errors	7-32
Errors Related to ILOM Watchdog State	7-32
Errors Related to Host Watchdog State	7-34

8 Using nvmeadm to Configure an NVMe Express Device

nvmeadm Command Overview	8-1
List NVMe Controllers	8-4
List NVMe Namespaces	8-5
List the Supported LBA Format	8-6
List NVMe Controller Log Pages	8-6
List NVMe Features of the Controller	8-7
Format All Namespaces on the Controller	8-8
Erase All Namespaces	8-8
Export an SSD Disk Configuration	8-9
Import an SSD Disk Configuration	8-9

9 Using raidconfig to Configure RAID

raidconfig Command Overview	9-1
raidconfig Features	9-1
raidconfig Requirements	9-2
raidconfig Command Syntax	9-2
Listing Controller, RAID and Disk Information	9-3
list Subcommand Overview	9-4
Display a Brief Listing of All Devices	9-6
Display a Brief Listing of a Device	9-7
Display a Detailed Listing of a Device	9-7
Creating and Deleting RAID Volumes	9-9
Create a RAID Volume	9-9
Delete a RAID Volume	9-11
Adding and Removing Disks and RAID Volumes	9-11
Add a Disk to a RAID Configuration	9-11
Remove a Disk from a RAID Volume	9-12
Add Spare Disks	9-12
Remove a Spare Disk or a RAID Volume	9-13
Modifying a RAID Volume or Controller	9-13
Modify a RAID Volume	9-14
Modify a Controller	9-14
Modify the BIOS Boot Target	9-15
Disable Auto Rebuild	9-15
Modify a RAID Volume Name	9-16
Enable or Disable JBOD Mode	9-16
Starting or Stopping a Task on a Disk or RAID	9-17
Executing Tasks on a Disk or RAID Volume	9-17
Start or Stop a Task on a Disk or RAID Volume	9-18
Restoring or Clearing a RAID Controller Configuration	9-19
Check to See If a Controller Configuration Exists	9-20
Restore a RAID Controller Configuration	9-20
Clear a RAID Controller Configuration	9-20
Exporting or Importing a RAID Volume Configuration	9-21
Export a RAID Volume Configuration	9-21
Import a RAID Volume Configuration	9-21
Creating RAID Volumes With Partial Disks	9-22
Guidelines for Using the RAID Volume Size Option	9-22
Disk Display	9-22
Partial Disk Properties in XML File	9-23
Create a RAID Volume with Partial Disks	9-23

10 Using ubiosconfig to Update the UEFI BIOS

ubiosconfig Command Overview	10-1
ubiosconfig Features	10-1
ubiosconfig and Service Processor Access	10-2
Obtaining SSL Certificates for TLS Access	10-2
Command Options for Accessing Oracle ILOM Over a Remote Network Connection	10-5
ubiosconfig Command Syntax	10-7
Export UEFI Settings to an XML File	10-9
Import UEFI BIOS Settings to a Server	10-10
Display Information on Changes to UEFI BIOS Settings	10-11
Cancel Pending Changes to UEFI BIOS Settings	10-11
Reset the UEFI BIOS Settings to Factory Default	10-11

11 Using zoningcli to Configure Zoning for the SPARC T3-1 Server

zoningcli Command Overview	11-1
List Zoning Information	11-2
Enable and Disable Zoning	11-2
Configure Zoning	11-2

12 CLI Tools Error Codes

Common Error Codes	12-1
biosconfig Error Codes	12-2
fwupdate Error Codes	12-3
hwmgmtcli Error Codes	12-4
ilomconfig Error Codes	12-4
nvmeadm Error Codes	12-6
raidconfig Error Codes	12-6
ubiosconfig Error Codes	12-8
zoningcli Error Codes	12-9

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/docs>.

Feedback

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

Change History

The following changes have been made to the documentation set.

- April 2017. Initial publication.
- July 2017. Updated *CLI Tools User's Guide* to add `--fail-without-interconnect` option to the `fwupdate update` command. Added information on Linux MMIO access settings for `fwupdate`. Added information on package requirements for InfiniBand adapters for `fwupdate`.
- September 2017. Updated *CLI Tools User's Guide* to list changed options for creating or modifying SNMP communities with Oracle ILOM 4.0.
- April 2018. Updated *CLI Tools User's Guide* to more accurately describe how `fwupdate` automatic mode works with XML metadata files.
- August 2018. Updated *CLI Tools User's Guide* to add new certificate checking options `--cert-dir` and `--no-cert-check` and description on obtaining remote SP certificates.
- February 2019. Updated *CLI Tools User's Guide* to add two new `fwupdate` targets `sp-bios-backup-firmware` and `sysfw-backup-firmware` for manual mode updates. Also added two new error codes and an updated description of the `-o` option for the `fwupdate list` command.

- February 2020. Updated *CLI Tools User's Guide* to add new `nvmeadm` subcommands, options and descriptions.
- May 2021. Updated *CLI Tools User's Guide* to update `fwupdate list` command examples. Added information about disabling secure boot to update firmware.
- March 2023. Updated *CLI Tools User's Guide* to add new IPMI interface options `-t` and `-T` descriptions to the `fwupdate`, `ilomconfig` and `ubiosconfig` commands. Added the `-Q` option description to the `fwupdate` command.

1

Oracle Server CLI Tools 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>.

The Oracle Hardware Management Pack download package available from My Oracle Support (MOS) includes support for select versions of Linux and Oracle Solaris (release 10, 11 and 11.1) operating systems. A cross-platform installer is available for all components. For more information on installing Oracle Hardware Management Pack components, refer to the .

Beginning with Oracle Hardware Management Pack 2.4, support for the Microsoft Windows operating system is **no longer available**.

Beginning with Oracle Solaris 11.2, Oracle Hardware Management Pack for Oracle Solaris has become an integrated component of the operating system. 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). For details, see <https://www.oracle.com/goto/ohmp/solarisdocs>.



Note:

This documentation applies to servers running supported versions of Linux and Oracle Solaris 10, 11 and 11.1 operating systems.

This guide includes information on the following Oracle Server CLI Tools:

Tool	Description	Link
<code>biosconfig</code>	Configure your server's BIOS CMOS settings and host boot order. This tool is only available for systems that <i>do not</i> have UEFI-enabled BIOS.	Using biosconfig to Update the BIOS
<code>fwupdate</code>	Update, query, and validate the firmware for Oracle server devices.	Using fwupdate to Update Firmware
<code>hwmgmtcli</code>	Get system information from the Oracle ILOM service processor.	Using hwmgmtcli to Display Hardware Information
<code>ilomconfig</code>	Manage Oracle ILOM configurations.	Using ilomconfig to Configure Oracle ILOM
<code>nvmeadm</code>	Modify the controller and device configuration on an NVM Express (NVMe) subsystem.	Using nvmeadm to Configure an NVM Express Device

Tool	Description	Link
<code>raidconfig</code>	Configure RAID volumes.	Using raidconfig to Configure RAID
<code>ubiosconfig</code>	Import and export your server's UEFI BIOS settings to an XML file. This tool is only available for systems that <i>have</i> UEFI-enabled BIOS.	Using ubiosconfig to Update the UEFI BIOS
<code>zoningcli</code>	Tool for Oracle SPARC T3-1 servers running Oracle Solaris OS. Enables you to configure systems that have the 16 disk backplane (SAS-2 expander) into two separated zones.	Using zoningcli to Configure Zoning for the SPARC T3-1 Server

For more information on other Oracle Hardware Management Pack features, see *Oracle Hardware Management Pack 2.4 Installation Guide* and *Oracle Hardware Management Pack 2.4 Server Management Agents User's Guide*.

For late-breaking issues and information about the CLI Tools, refer to the *Oracle Hardware Management Pack 2.4 Release Notes*.

2

Host-to-ILOM Interconnect

With Oracle ILOM 3.0.12 and later, a Host-to-ILOM Interconnect communication channel is available to enable you to communicate locally with Oracle ILOM from the host operating system (OS) without the use of a network management connection (NET MGT) to the server's service processor.



Note:

The Oracle Hardware Management Pack refers to this feature as Host-to-ILOM Interconnect. The Oracle ILOM interface refers to this feature as Local Host Interconnect or LAN-over-USB.

The Host-to-ILOM Interconnect is available on the latest Oracle servers and the can provide a more reliable and potentially faster data transfer rate for Oracle Hardware Management Pack CLI tools than traditional KCS interfaces.

Accessing any service processor over a remote network connection, instead of the Host-to-ILOM interconnect, is also available for certain Oracle Hardware Management commands. This method additionally requires that you provide a host name or IP address and user account credentials of the service processor on which the command is being executed.

For more information on enabling the Host-to-ILOM Interconnect, refer to [Enabling the Host-to-ILOM Interconnect](#) in *Oracle Hardware Management Pack 2.4 Installation Guide*.

3

CLI Tools Command Syntax and Conventions

The following information is covered in this section.

- [CLI Tools Command Syntax](#)
- [CLI Tools Device-Naming Convention](#)

CLI Tools Command Syntax

Most CLI tools commands conform to one of the following two command syntax formats:

- `command [option]`
- `command subcommand target [option]`



Note:

The `biosconfig` tool does not conform to the above syntax. See [Using biosconfig to Update the BIOS](#) for more information.

The following table describes the command fields.

Command Field	Description	Examples
<code>command</code>	The action that you want to perform. Identifies that CLI tool that you are using. Consists of lower-case letters only.	<code>biosconfig</code> , <code>fwupdate</code> , <code>raidconfig</code> , <code>ilomconfig</code>
<code>subcommand</code>	Further defines the task to be performed by the <code>command</code> . Generally used as verbs. Consists of lower-case letters, hyphens, or the underscore character. The subcommand is not required when the <code>--version</code> or <code>--help</code> option is used immediately following the command.	<code>list</code> , <code>update</code> , <code>reset</code>
<code>target</code>	Describes the object or target that is being acted upon by the subcommand. Application specific.	<code>all</code> , <code>disk</code> , <code>expander</code> , <code>bridge</code> , <code>controller</code> , <code>user</code> , <code>snmp-community</code>

Command Field	Description	Examples
<i>option</i>	<p>Modifies the command or subcommand and can be optional or mandatory depending on the command or subcommand.</p> <p>There are long and short options that have identical functionality and are provided for ease of use:</p> <p>Short-option is a hyphen followed by a single letter.</p> <p>Long-option is two hyphens followed by a string.</p>	<p>-n or --device_name</p> <p>-f or --filename</p> <p>-r or --reset</p>

The following options apply to all CLI Tools commands.

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	Confirms operation. Does not prompt user for confirmation on the operation when running.

When using a command option and its corresponding value or device name, you can use an equal sign (=) or a space as shown in the following examples:

- Using a command with spaces:



```
raidconfig create raid -c c2 --raid-level 1 --number-disks 2
```
- Using a command with equal signs (=):

```
raidconfig create raid -c=c2 --raid-level=1 --number-disks=2
```

CLI Tools Device-Naming Convention

The following table lists device names are used with the CLI Tools commands. The character identifier represents all of the nodes that make up the device.

Identifier	Description
c	The controller, followed by a unique logical ID.
r	The RAID Volume (logical disk), followed by a logical ID name of the volume or disk.
d	The disk, followed by the physical disk logical ID name.
x	The expander, followed by the unique expander logical ID name.
j	The chassis, followed by the unique chassis logical ID name.

Identifier	Description
sp_bios	A system service processor. <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> Note: Use this device identifier with Oracle Hardware Management Pack 2.3.5 and earlier.</p> </div>
sp	A system service processor. <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> Note: Use this device identifier with Oracle Hardware Management Pack 2.3.6 and later.</p> </div>

All integers used to represent the device are 0 based. Disks are represented by logical ID names assigned by the tool at initialization. The disks are sorted by expander and slot ID to create unique numerical identifiers.

The following are examples of device names:

- c1 – Controller 1
- c1d2 – Disk with a logical ID 2 on controller 1
- c2r1 – RAID 1 on controller 2

Multiple devices can be listed together in a comma-separated list, for example:
device1,device2,device3.

The following example shows a `raidconfig` command for creating a RAID volume with three disks:

```
raidconfig create --disks c1d2,c1d4,c1d5 --level 1
```

The following example shows an implementation of the disk-naming scheme.

ID	Brand	Model	Chassis	Slot	Type	Media	Size (GB)	Firmware Revision
c1d0	SEAGATE	ST373455SSUN72G	0	0	sas	HDD	73	0791
c1d1	SEAGATE	ST35000N	0	1	sata	HDD	500	3AZQ
c1d2	SEAGATE	ST373455SSUN72G	0	2	sas	HDD	73	0B92
c1d3	SEAGATE	ST373455SSUN72G	0	3	sas	HDD	73	0B92
c1d4	SEAGATE	ST35000N	0	4	sata	HDD	500	3AZQ
c1d5	SEAGATE	ST35000N	0	5	sata	HDD	500	3AZQ
c1d6	SEAGATE	ST35000N	0	6	sata	HDD	500	3AZQ
c1d7	SEAGATE	ST373455SSUN72G	0	7	sas	HDD	73	0B92
c1d8	SEAGATE	ST373455SSUN72G	0	8	sas	HDD	73	0B92
c1d9	SEAGATE	ST373455SSUN72G	0	9	sas	HDD	73	0B92
c1d10	SEAGATE	ST35000N	0	10	sata	HDD	500	3AZQ

c1d11 0B92	SEAGATE	ST373455SSUN72G	0	11	sas	HDD	73
c1d12 0B92	SEAGATE	ST373455SSUN72G	0	12	sas	HDD	73
c1d13 0B92	SEAGATE	ST373455SSUN72G	0	13	sas	HDD	73
c1d14 0B92	SEAGATE	ST373455SSUN72G	0	14	sas	HDD	73
c1d15 0B92	SEAGATE	ST373455SSUN72G	0	15	sas	HDD	73
c1d16 0B92	SEAGATE	ST373455SSUN72G	0	16	sas	HDD	73
c1d17 0B92	SEAGATE	ST373455SSUN72G	0	17	sas	HDD	73
c1d18 0B92	SEAGATE	ST373455SSUN72G	0	18	sas	HDD	73
c1d19 0B92	SEAGATE	ST373455SSUN72G	0	19	sas	HDD	73
c1d20 3AZQ	SEAGATE	ST35000N	0	20	sata	HDD	500
c1d21 3AZQ	SEAGATE	ST35000N	0	21	sata	HDD	500
c1d22 3AZQ	SEAGATE	ST35000N	0	22	sata	HDD	500
c1d23 3AZQ	SEAGATE	ST35000N	0	23	sata	HDD	500
c1d24 0791	SEAGATE	ST373455SSUN72G	1	0	sas	HDD	73
c1d25 3AZQ	SEAGATE	ST35000N	1	1	sata	HDD	500
c1d26 0791	SEAGATE	ST373455SSUN72G	1	3	sas	HDD	73
c1d27 3AZQ	SEAGATE	ST35000N	1	4	sata	HDD	500
c1d28 0791	SEAGATE	ST373455SSUN72G	1	5	sas	HDD	73
c1d29 3AZQ	SEAGATE	ST35000N	1	6	sata	HDD	500
c1d30 0791	SEAGATE	ST373455SSUN72G	1	7	sas	HDD	73
c1d31 0791	SEAGATE	ST373455SSUN72G	1	8	sas	HDD	73
c1d32 0791	SEAGATE	ST373455SSUN72G	1	9	sas	HDD	73
c1d33 0791	SEAGATE	ST373455SSUN72G	1	10	sas	HDD	73
c1d34 0791	SEAGATE	ST373455SSUN72G	1	11	sas	HDD	73
c1d35 3AZQ	SEAGATE	ST35000N	1	12	sata	HDD	500
c1d36 0791	SEAGATE	ST373455SSUN72G	1	13	sas	HDD	73
c1d37 0791	SEAGATE	ST373455SSUN72G	1	14	sas	HDD	73
c1d38 3AZQ	SEAGATE	ST35000N	1	15	sata	HDD	500
c1d39 0791	SEAGATE	ST373455SSUN72G	1	16	sas	HDD	73
c1d40 0791	SEAGATE	ST373455SSUN72G	1	17	sas	HDD	73
c1d41	SEAGATE	ST35000N	1	18	sata	HDD	500

3AZQ									
c1d42	SEAGATE	ST35000N	1	19	sata	HDD	500	3AZQ	
c1d43	SEAGATE	ST35000N	1	20	sata	HDD	500	3AZQ	
c1d44	SEAGATE	ST35000N	1	21	sata	HDD	500	3AZQ	
c1d45	SEAGATE	ST35000N	1	22	sata	HDD	500	3AZQ	
c1d46	SEAGATE	ST35000N	1	23	sata	HDD	500	3AZQ	

4

Using biosconfig to Update the BIOS

`biosconfig` configures the BIOS CMOS settings, host boot order, and some service processor settings.



Note:

The `biosconfig` tool is used to configure system BIOS (also called "legacy BIOS") on supported Oracle x86 servers. Servers that support UEFI BIOS must use the `ubiosconfig` tool. See [Using ubiosconfig to Update the UEFI BIOS](#).

For a list of the tools and the systems that support them, refer to:

<https://www.oracle.com/goto/ohmp>

`biosconfig` allows you to manipulate BIOS configurations from the OS command line. The configuration files and command-line interfaces are compatible with the Oracle Solaris and Linux OSes.

- [biosconfig Command Overview](#)
- [biosconfig for Oracle Solaris OS](#)
- [Viewing biosconfig Command Options and Version Information](#)
- [Configuring the Device Boot Order](#)
- [Configuring the BIOS CMOS](#)
- [Commands That Produce Unrelated, Innocuous, Extra Output](#)

biosconfig Command Overview

This section covers the following information:

- [biosconfig Requirements](#)
- [biosconfig Device Terminology](#)
- [Editing XML Files](#)
- [biosconfig Command Syntax](#)

biosconfig Requirements

- You must run `biosconfig` as root because it needs to use drivers that are in read- and write-protected physical address space.
- Close all other applications and quiesce your system before running `biosconfig`.

- Linux versions of `biosconfig` depend on access to `/dev/nvram` to guarantee serialized access to the CMOS.

Red Hat Enterprise Linux 4 distributions do not include the device by default. Red Hat Enterprise Linux 5 and SUSE Linux Enterprise Server distributions do include the device by default.

To use `/dev/nvram`, the driver needs to be compiled into the kernel (or loaded as a module), and `/dev/nvram` must exist (root can create it using `mknod /dev/nvram c 10 144`).

biosconfig Device Terminology

The following notes explain how `biosconfig` describes devices:

- Floppy refers to whatever the BIOS considers a removable device. For example, this could be a USB flash drive.
- A USB flash drive bigger than 512 MB is referred to as a disk.
- A USB/CD-ROM is classified as a CD and not a removable device.
- PXE means a bootable network device. For example, this might be an Ethernet controller or an InfiniBand interface that has boot support in its expansion ROM.
- [Device Name Examples](#)

Device Name Examples

The device name examples listed in the following table are used in XML file output in this chapter.

Output Text	Description of Hardware
SATA:3M-MRVLRD 200254-01SUN24G 0801	Flash mini-DIMM SATA (which is disk-like)
USB:Port1:Memorex DVD+-RAM 510L	USB DVD drive (which is CD-like)
USB:Port0:SanDisk Cruzer Contour	1 GB USB flash drive (which is disk-like)
IB:Slot2.F0:PXE:MLNX HCA IB 1.9.972	InfiniBand PXE (which is network-like)
PXE:IBA GE Slot 00C8 v1324	On-board GigEthernet NIC (which is a network interface)

Editing XML Files

`biosconfig` enables you to configure settings across multiple similar servers using a common XML configuration file. However, if the configuration that is being modified includes a peripheral or component that is not on both systems, then you need to customize the XML file. The BIOS firmware of systems you are exporting from or importing to does not have to be at the same version.

 **Note:**

The XML tag definitions are determined by the current system BIOS. These values can vary by system type and it is not recommended that you use the XML file to update the BIOS configuration across different system types.

The `biosconfig` command can be used to get current configuration settings or set configuration settings. When used to get configuration settings, `biosconfig` generates XML output showing the configuration. When used to set configuration settings, `biosconfig` reads XML input describing the configuration settings.

 **Caution:**

Do not use `biosconfig` to change BIOS settings that are not visible in the normal BIOS setup menu.

To use `biosconfig`, you must have a working knowledge of XML file editing. The process of editing the BIOS includes using `biosconfig` to do the following tasks:

1. To obtain the BIOS configuration settings in XML, type:

```
# biosconfig -get_option filename.xml
```

If an XML file name is specified with the `get` command, the BIOS configuration is saved to the XML file. If an XML file is not specified, the output is written to the terminal.

2. Review the XML file and modify it, as required.

You can modify the XML files in a editor of your choice, such as `vi`.

3. To implement the changes, type:

```
# biosconfig -set_option filename.xml
```

You can use the same XML file to modify multiple systems of the same type.

biosconfig Command Syntax

The `biosconfig` command uses the following syntax:

```
biosconfig [-v] option [filename.xml]
```

When a command fails, it returns one of the failure codes listed in [biosconfig Error Codes](#).

The following table lists the available `biosconfig` options and their descriptions.

Option	Description
<code>-get_version</code>	Get version of this tool.
<code>-get_boot_order</code>	Get the boot devices list.
<code>-set_boot_order</code>	Set the boot devices list.
<code>-set_boot_override</code>	Set the first boot device for the next boot.

Option	Description
<code>-get_bios_settings</code>	Get setup configuration from BIOS.
<code>-set_bios_settings</code>	Get setup configuration to BIOS ROM.
<code>-get_CMOS_dump</code>	Get 256 bytes CMOS of set up data from BIOS.
<code>-set_CMOS_dump</code>	Set 256 bytes of CMOS set up data to BIOS.
<code>-v</code>	Verbose mode. On some operations, this might provide additional information regarding operational status. Verbose mode is only valid if an XML input or output filename is provided.

The following table lists examples of how the `-get` and `-set` command options affect input and output.

Command	Description
<code>biosconfig -get_version</code>	Outputs to screen.
<code>biosconfig -get_version filename.xml</code>	Outputs to <i>filename.xml</i> .
<code>biosconfig -get_version > filename.xml</code>	Outputs to <i>filename.xml</i> .
<code>biosconfig -get_version some-command</code>	Pipes the output to another command.
<code>biosconfig -set_bios_settings</code>	Takes input from standard in.
<code>biosconfig -set_bios_settings filename.xml</code>	Takes input from <i>filename.xml</i> .
<code>biosconfig -set_bios_settings < filename.xml</code>	Takes input from <i>filename.xml</i> .

 **Note:**

In the output examples in this chapter, all white space outside the XML elements, such as indentation, is optional. For example, see the output in [Make a Persistent Change to Boot Order](#).

biosconfig for Oracle Solaris OS

The BIOS configuration tool for Oracle Solaris OS (`biosconfig`) is a utility that runs on the host system and configures the host's BIOS CMOS settings, host boot order, and some service processor settings.

Oracle Solaris OS `biosconfig` consists of an Oracle Solaris OS `biosdrv` driver and the `biosconfig` application.

Viewing biosconfig Command Options and Version Information

This section covers the following information:

- [View biosconfig Command Options](#)
- [View biosconfig Version Information](#)

View biosconfig Command Options

- To view the help output, execute the biosconfig command without arguments. Type:

```
# biosconfig
```

For example:

```
# biosconfig
Copyright (C) SUN Microsystems 2009.
BIOSconfig Utility Version 2.2.5
Build Date: Jan 11 2010
Build Time: 01:22:05
```

```
BIOSconfig Specification Version 2.4
```

```
Usage: biosconfig [-v] option [filename]
Example: biosconfig -get_version output.xml
```

```
[-v] Verbose on. Only valid if a xml input/output filename is provided
[Filename] Name of the XML output (or input) file for get (or set)
command (optional).
get commands will output to the console if the filename
is not provided
set commands will get input from the console if the filename
is not provided
```

Available options (Required):

```
-get_version Get version of this tool
-get_boot_order Get the BOOT Devices list
-set_boot_order Set the BOOT Devices list
-get_bios_settings Get setup configuration from BIOS
-set_bios_settings Set setup configuration to BIOS ROM
-get_CMOS_dump Get 256 bytes CMOS setup data from BIOS
-set_CMOS_dump Set 256 bytes of CMOS setup data to BIOS
```

View biosconfig Version Information

- To view version information and save it to an XML file, type:

```
# biosconfig -get_version filename.xml
```

For example:

```
# biosconfig -get_version ver.xml
```

```
Copyright (C) SUN Microsystems 2009.
BIOSconfig Utility Version 2.1
Build Date: Jul 16 2009
Build Time: 15:55:12
```


BIOSconfig Specification Version 2.4

Success

If you do not include the *filename* option in the command, the version information is displayed on the screen.

The following is an example of how the version information is stored in an XML file.

```
<?xml version="1.0" encoding="UTF-8"?>
<BIOSCONFIG>
  <BIOSCONFIG_VERSION>2.1</BIOSCONFIG_VERSION>
  <SPEC_VERSION>2.4</SPEC_VERSION>
  <SP_NETWORK_CONFIG>
    <DISCOVERY></DISCOVERY>
    <IP></IP>
    <NETMASK></NETMASK>
    <GATEWAY></GATEWAY>
  </SP_NETWORK_CONFIG>
  <PASSWORD_CONFIG>
    <PASSWORD></PASSWORD>
  </PASSWORD_CONFIG>
  <BOOT_ORDER_OVERRIDE>
    <HELP_STRING>FIRST=Choose one of: pxe, cdrom, disk,
floppy, bios, none</HELP_STRING>
    <FIRST></FIRST>
    <HELP_STRING>CLEAR_CMOS=Choose Yes, No or leave it
empty, em....</HELP_STRING>
    <CLEAR_CMOS></CLEAR_CMOS>
  </BOOT_ORDER_OVERRIDE>
  <BOOT_DEVICE_PRIORITY>
    <B0>
      <DEVICE_NAME></DEVICE_NAME>
      <PCI-B-D-F></PCI-B-D-F>
    </B0>
  </BOOT_DEVICE_PRIORITY>
</BIOSCONFIG>
```

Configuring the Device Boot Order

During BIOS power-on self-test (POST), BIOS scans the hardware and accumulates a list of bootable devices. That list is then presented as a boot list, which is the ordered list of bootable devices.

`biosconfig` enables you to configure the first device to boot at the next reboot or to configure the entire boot order. `biosconfig` does this by reading the boot-related tables that the BIOS stores in NVRAM and then manipulating the contents of CMOS where the boot order is stored.

- [Methods for Changing the Boot List](#)
- [Set the First Boot Device for the Next Boot](#)
- [Make a Persistent Change to Boot Order](#)
- [Change Boot Order Based on the PCI Bus, Device, or Function](#)

Methods for Changing the Boot List

The boot list can be changed in any of the following ways:

- Change the order in BIOS setup utility.
- Reorder the categories using the IPMI bootflags that the SP offers to the compatible BIOS during POST. The default priority order for categories is CD/DVD, disk, removable, and network.
- Change the boot order using `biosconfig`. This manipulates the contents of CMOS and the BIOS boot block structures stored in NVRAM, which is a dedicated part of the BIOS ROM.

This section contains instructions for changing the boot order using `biosconfig`.

Note:

This boot list changes dynamically when devices such as disk drives, USB devices, and PCIe cards are installed and removed. The boot list also changes when `javaConsole` floppy and CD redirection is started and stopped.

Set the First Boot Device for the Next Boot

This procedure shows how to set the first boot device for the next boot only. To change the boot device for successive boots, see [Make a Persistent Change to Boot Order](#).

Here is an example of using the `-set_boot_override` command that specifies the first boot device as the PXE server on only the next boot:

1. To create an XML file containing the current boot order of your system, type:

```
# biosconfig -get_boot_order filename.xml
```

2. Edit XML text so that the device that you want to boot first is between the `<FIRST>` tags.

In this example, the PXE device is the first boot device.

The following is an example of the resulting XML file.

```
<?xml version="1.0" encoding="UTF-8"?>
<BIOSCONFIG>
  <BIOSCONFIG_VERSION>2.1</BIOSCONFIG_VERSION>
  <SPEC_VERSION>2.4</SPEC_VERSION>
  <SP_NETWORK_CONFIG>
    <DISCOVERY></DISCOVERY>
    <IP></IP>
    <NETMASK></NETMASK>
    <GATEWAY></GATEWAY>
  </SP_NETWORK_CONFIG>
  <PASSWORD_CONFIG>
    <PASSWORD></PASSWORD>
  </PASSWORD_CONFIG>
  <BOOT_ORDER_OVERRIDE>
    <HELP_STRING>FIRST=Choose one of: pxe, cdrom, disk,
floppy, bios, none</HELP_STRING>
    <FIRST>pxe</FIRST>
  </BOOT_ORDER_OVERRIDE>
</BIOSCONFIG>
```

```

    <HELP_STRING>CLEAR_CMOS=Choose Yes, No or leave it
empty, ....</HELP_STRING>
    <CLEAR_CMOS></CLEAR_CMOS>
  </BOOT_ORDER_OVERRIDE>
  <BOOT_DEVICE_PRIORITY>
  <Boot_Device_01>
    <DEVICE_NAME>USB:Port1:Memorex DVD+-RAM 510L v1</DEVICE_NAME>
  </Boot_Device_01>
  <Boot_Device_02>
    <DEVICE_NAME>SATA:3M-MRVLRD 200254-01SUN24G 0801</DEVICE_NAME>
  </Boot_Device_02>
  <Boot_Device_03>
    <DEVICE_NAME>USB:Port0:SanDisk Cruzer Contour</DEVICE_NAME>
  </Boot_Device_03>
  <Boot_Device_04>
    <DEVICE_NAME>IB:Slot2.F0:PXE:MLNX HCA IB 1.9.972 (PCI 07:00.
</DEVICE_NAME>
    <PCI-B-D-F>07,00,00</PCI-B-D-F>
  </Boot_Device_04>
  <Boot_Device_05>
    <DEVICE_NAME>PXE:IBA GE Slot 00C8 v1324</DEVICE_NAME>
    <PCI-B-D-F>00,19,00</PCI-B-D-F>
  </Boot_Device_05>
  </BOOT_DEVICE_PRIORITY>
</BIOSCONFIG>

```

3. To set the boot order, type:

```
# biosconfig -set_boot_override filename.xml
```

Make a Persistent Change to Boot Order

To make a persistent change to the boot order, modify the order of devices between the `BOOT_DEVICE_PRIORITY` tags of the XML file.

The following example shows an XML file from a Sun Blade X6275 server module (which has a built-in bootable InfiniBand interface) set to optimal defaults with a 1-GBYTE USB flash, a USB CD, and a dual Gig-Ethernet Express Module plugged in.

1. To create an XML file containing the current boot order of your system, type:

```
# biosconfig -get_boot_order filename.xml
```

The following is an example of the output of the XML file:

```

<?xml version="1.0" encoding="UTF-8"?>
<BIOSCONFIG>
  <BIOSCONFIG_VERSION>2.1</BIOSCONFIG_VERSION>
  <SPEC_VERSION>2.4</SPEC_VERSION>
  <SP_NETWORK_CONFIG>
    <DISCOVERY></DISCOVERY>
    <IP></IP>
    <NETMASK></NETMASK>
    <GATEWAY></GATEWAY>
  </SP_NETWORK_CONFIG>
  <PASSWORD_CONFIG>
    <PASSWORD></PASSWORD>
  </PASSWORD_CONFIG>
  <BOOT_ORDER_OVERRIDE>
    <HELP_STRING>FIRST=Choose one of: pxe, cdrom, disk,
floppy, bios, none</HELP_STRING>
    <FIRST></FIRST>

```

```

    <HELP_STRING>CLEAR_CMOS=Choose Yes, No or leave it
empty, ....</HELP_STRING>
    <CLEAR_CMOS></CLEAR_CMOS>
  </BOOT_ORDER_OVERRIDE>
  <BOOT_DEVICE_PRIORITY>
  <Boot_Device_01>
    <DEVICE_NAME>USB:Port1:Memorex DVD+-RAM 510L v1</DEVICE_NAME>
  </Boot_Device_01>
  <Boot_Device_02>
    <DEVICE_NAME>SATA:3M-MRVLRD 200254-01SUN24G 0801</DEVICE_NAME>
  </Boot_Device_02>
  <Boot_Device_03>
    <DEVICE_NAME>USB:Port0:SanDisk Cruzer Contour</DEVICE_NAME>
  </Boot_Device_03>
  <Boot_Device_04>
    <DEVICE_NAME>IB:Slot2.F0:PXE:MLNX HCA IB 1.9.972 (PCI 07:00.
</DEVICE_NAME>
    <PCI-B-D-F>07,00,00</PCI-B-D-F>
  </Boot_Device_04>
  <Boot_Device_05>
    <DEVICE_NAME>PXE:IBA GE Slot 00C8 v1324</DEVICE_NAME>
    <PCI-B-D-F>00,19,00</PCI-B-D-F>
  </Boot_Device_05>
  </BOOT_DEVICE_PRIORITY>
</BIOSCONFIG>

```

2. Edit the device names displayed between the <DEVICE_NAME> tags so that the devices are listed in the desired boot order.
3. To set the boot order, type:

```
# biosconfig -set_boot_order filename.xml
```

Change Boot Order Based on the PCI Bus, Device, or Function

The `biosconfig` command can alter the boot order based on the PCI bus, device, or function if the boot order list contains that information.

1. To create an XML file containing the current boot order of your system, type:

```
# biosconfig -get_boot_order filename.xml
```
2. Edit the devices listed between the <PCI-B-D-F> tags so that they are in the desired order.

For example:

```

<BOOT_DEVICE_PRIORITY>
<Boot_Device_01>
  <DEVICE_NAME>PXE:IBA GE Slot 00C8 v1324</DEVICE_NAME>
  <PCI-B-D-F>00,19,00</PCI-B-D-F>
</Boot_Device_01>
<Boot_Device_02>
  <DEVICE_NAME>IB:Slot2.F0:PXE:MLNX HCA IB 1.9.972 (PCI 07:00.</DEVICE_NAME>
  <PCI-B-D-F>07,00,00</PCI-B-D-F>
</Boot_Device_02>
<Boot_Device_03>
  <DEVICE_NAME>USB:Port1:Memorex DVD+-RAM 510L v1</DEVICE_NAME>
</Boot_Device_03>
<Boot_Device_04>
  <DEVICE_NAME>USB:Port0:SanDisk Cruzer Contour</DEVICE_NAME>
</Boot_Device_04>

```

```
<Boot_Device_05>  
  <DEVICE_NAME>SATA:3M-MRVLRD 200254-01SUN24G 0801</DEVICE_NAME>  
</Boot_Device_05>  
</BOOT_DEVICE_PRIORITY>
```

3. To set the boot order, type:

```
# biosconfig -set_boot_order filename.xml
```

Configuring the BIOS CMOS

The BIOS configuration information is stored in the CMOS memory in the host's chipset. You can use `biosconfig` to modify these settings with a program on the host OS. Alternatively, you can configure many of the CMOS settings through the BIOS setup interface at BIOS POST.

`biosconfig` configures the BIOS CMOS settings using two methods:

- Copying and using a golden (known reliable) image
- Controlling each setting individually
- [Capture the BIOS CMOS Golden Image](#)
- [Apply the BIOS CMOS Golden Image](#)
- [Configuring Individual CMOS Settings](#)

Capture the BIOS CMOS Golden Image

The BIOS configuration consists of the contents of the CMOS and the boot tables in the NVRAM. The command `biosconfig get_cmos_dump` captures the 256 bytes of CMOS, but it does not gather the boot table information from NVRAM. So this command might not capture the boot-order information, unless the bootable I/O configurations for the source and destination machines are the same.

1. To generate a golden (known reliable) CMOS image, use the BIOS Setup Utility to configure the BIOS settings.
2. To capture the 256 bytes of CMOS containing the configuration information, type:

```
# biosconfig -get_cmos_dump filename.xml
```

The following display shows an example of the output.

```
Copyright (C) SUN Microsystems 2009.  
BIOSconfig Utility Version 2.1  
Build Date: Jul 16 2009  
Build Time: 15:55:12  
BIOSconfig Specification Version 2.4  
Success
```

The following example shows an XML file containing the CMOS configuration information:

```
<BIOSCONFIG>  
  <BIOSCONFIG_VERSION>2.1</BIOSCONFIG_VERSION>  
  <SPEC_VERSION>2.4</SPEC_VERSION>  
  <SP_NETWORK_CONFIG>  
    <DISCOVERY></DISCOVERY>  
    <IP></IP>  
    <NETMASK></NETMASK>
```

```

    <GATEWAY></GATEWAY>
  </SP_NETWORK_CONFIG>
  <PASSWORD_CONFIG>
    <PASSWORD></PASSWORD>
  </PASSWORD_CONFIG>
  <BOOT_ORDER_OVERRIDE>
    <HELP_STRING>FIRST=Choose one of: pxe, cdrom, disk, floppy,
bios, none</HELP_STRING>
    <FIRST></FIRST>
    <HELP_STRING>CLEAR_CMOS=Choose Yes, No or leave it empty,
</HELP_STRING>
    <CLEAR_CMOS></CLEAR_CMOS>
  </BOOT_ORDER_OVERRIDE>
  <BOOT_DEVICE_PRIORITY>
    <B0>
      <DEVICE_NAME></DEVICE_NAME>
      <PCI-B-D-F></PCI-B-D-F>
    </B0>
  </BOOT_DEVICE_PRIORITY>
  <CMOS_DUMP>
  <OFFSET_00>00.00.00.00.00.00.00.00.00.00.00.00.00.00.00.00.</OFFSET_00>
  <OFFSET_10>00.30.00.30.0E.80.02.FF.FF.00.00.00.00.00.00.</OFFSET_10>
  <OFFSET_20>00.00.00.00.00.00.00.00.00.30.47.47.47.47.04.3A.</OFFSET_20>
  <OFFSET_30>FF.FF.20.85.90.F7.07.00.00.03.00.17.00.00.1F.3A.</OFFSET_30>
  <OFFSET_40>00.00.00.00.00.00.00.00.00.00.00.00.00.00.00.</OFFSET_40>
  <OFFSET_50>00.00.FF.00.13.00.00.01.80.30.30.30.30.00.00.</OFFSET_50>
  <OFFSET_60>EF.40.41.42.43.44.45.46.47.08.09.0A.18.00.00.0B.</OFFSET_60>
  <OFFSET_70>00.03.0C.0D.0E.0F.10.11.00.00.00.00.12.13.14.15.</OFFSET_70>
  <OFFSET_80>11.24.26.06.46.14.00.16.02.00.F8.23.C8.17.20.07.</OFFSET_80>
  <OFFSET_90>18.20.19.1A.1B.1C.1D.9E.DF.9E.DE.21.02.03.04.05.</OFFSET_90>
  <OFFSET_A0>06.07.08.09.EA.2B.0B.0B.0B.4B.00.01.0F.00.0C.00.</OFFSET_A0>
  <OFFSET_B0>00.00.00.00.10.32.54.76.10.32.54.76.14.00.00.00.</OFFSET_B0>
  <OFFSET_C0>00.46.BC.00.00.00.00.00.00.80.C0.10.42.F9.FF.FF.</OFFSET_C0>
  <OFFSET_D0>83.00.80.9C.DE.1F.40.02.FA.52.55.E0.F1.F3.E7.FF.</OFFSET_D0>
  <OFFSET_E0>7C.00.01.04.00.00.05.04.03.04.00.02.07.02.17.00.</OFFSET_E0>
  <OFFSET_F0>17.03.01.05.08.01.03.04.00.03.00.09.01.00.05.00.</OFFSET_F0>
  </CMOS_DUMP>
</BIOSCONFIG>

```

 **Note:**

The data between the <CMOS_DUMP> element tags contains raw CMOS data.

Apply the BIOS CMOS Golden Image

You can apply the golden image to identical hardware by copying the golden image from your source system to a destination system with the same BIOS revision, as shown using `-set_cmos_dump`.

1. Copy the filename.xml image from your source system to a destination system.
2. To apply the golden image to the destination system, type:

```
# biosconfig -set_cmos_dump filename.xml
```

Copyright (C) SUN Microsystems 2009.
BIOSconfig Utility Version 2.1

```
Build Date: Jul 16 2009
Build Time: 15:55:12

BIOSconfig Specification Version 2.4

Processing Input BIOS Data....

Success
```

Configuring Individual CMOS Settings

`biosconfig` provides two commands to manage individual CMOS settings:

- `biosconfig -get_bios_settings`
Gets CMOS settings from the platform.
- `biosconfig -set_bios_settings`
Sets CMOS settings on the platform.

Options to use with the commands:

1. Use `-get_bios_settings filename.xml` to generate an XML file that describes the current settings.
2. Edit that XML file to change the settings.
3. Use `-set_bios_settings filename.xml` to apply the settings to CMOS.

You can provide a subset of the XML file to include only the settings that you want to change with the `-get_bios_settings` option. The XML file must be valid, so you must remove entire option sets from the XML file.

Note:

Values for the settings vary depending on your server type. `biosconfig` reads the host's BIOS image and the platform's CMOS to find the setup questions (the strings displayed in BIOS setup), the optimal default values, the current settings, and the permitted settings. The XML file structure matches the menu hierarchy in BIOS setup.

The names in the output XML file match the names in the setup menus; the only difference is that the spaces are replaced with underscores (`_`). For example, the Quick Boot entry in the Boot Settings Configuration submenu in the Boot menu of BIOS setup is specified like this:

```
<BIOSCONFIG>
<SETUP_CONFIG>
<Boot>
<Boot_Settings_Configuration>
<Quick_Boot>
```

- [Static and Dynamic CMOS Settings](#)
- [Configure a Static CMOS Setting](#)
- [Configure a Dynamic Setting](#)

Static and Dynamic CMOS Settings

There are two types of CMOS settings: static and dynamic. Static settings are human readable and dynamic settings are numerical. The following settings are determined at runtime by the BIOS:

- The value in CMOS
- The behavior determined by that value
- The BIOS setup strings displayed

Configure a Static CMOS Setting

The following procedure describes how to set static CMOS settings. The XML samples shown are subsets of the output XML file.

1. To get the CMOS settings, type:

```
# biosconfig -get_bios_settings filename.xml
```

2. View the XML file.

For example:

```
<BIOSCONFIG>
  <SETUP_CONFIG>
    <Boot>
      <Boot_Settings_Configuration>
        <Quick_Boot>
          <HELP_STRING>Allows BIOS to skip certain....
        </HELP_STRING>
        <DEFAULT_OPTION>Enabled</DEFAULT_OPTION>
        <SELECTED_OPTION>Enabled</SELECTED_OPTION>
        <OPTION-0>Disabled</OPTION-0>
        <OPTION-1>ENabled</OPTION-1>
      </Quick_Boot>
      <Onboard_IB_gPXE_boot_first_>
        <HELP_STRING>Set Onboard Infiniband gPXE ....
      </HELP_STRING>
      <DEFAULT_OPTION>Disabled</DEFAULT_OPTION>
      <SELECTED_OPTION>Disabled</DEFAULT_OPTION>
      <OPTION-0>Disabled</OPTION-0>
      <OPTION-1>Enabled</OPTION-2>
    </Onboard_IB_gPXE_boot_first_>
  </Boot_Settings_Configuration>
</Boot>
</SETUP_CONFIG>
</BIOSCONFIG>
```

3. Modify the value in the <SELECTED_OPTION> tags, as needed.

The options listed below the <SELECTED_OPTION> tags display the available values.

For example, the options for the Quick Boot setting are Disabled and Enabled

4. To set the static CMOS values, type:

```
# biosconfig -set_bios_settings filename.xml
```


Configure a Dynamic Setting

`biosconfig` cannot retrieve the strings and the mapping between the values in CMOS. This behavior is BIOS dependent; the ability to retrieve this information depends on the BIOS revision and the platform type.

To configure or export dynamic settings, you need to discover the setting that you wish to use by following these steps:

1. Enter the BIOS Setup utility.
2. Configure the settings manually and save the configuration.
3. To examine the resulting XML output to find the value that the BIOS is using for the setting you wish to specify, type:

```
# biosconfig -get_bios_settings filename.xml
```

This is an example of a dynamic CMOS setting as displayed in the XML file:

```
<BIOSCONFIG>
  <SETUP_CONFIG>
    <Boot>
      <Option_ROM_Enable>
      <NET0_Option_ROM_>
      <HELP_STRING>This Option enables execut....
    </HELP_STRING>
      <DEFAULT_OPTION> 0000 </DEFAULT_OPTION>
      <SELECTED_OPTION> 0000 </SELECTED_OPTION>
      <OPTION_RANGE> 0000 - 0001 </OPTION_RANGE>
      <OPTION-0>Not Available</OPTION-0>
    </NET0_Option_ROM_>
    </Option_ROM_Enable>
  </Boot>
</SETUP_CONFIG>
</BIOSCONFIG>
```

In the preceding code, there are no string-to-value mappings offered by the `biosconfig` output.

4. To set the BIOS configuration, type:

```
biosconfig -set_bios_settings filename.xml
```

Use this XML file to configure dynamic CMOS settings on machines of the same model.

Commands That Produce Unrelated, Innocuous, Extra Output

The following is a known issue with `biosconfig`.

Some commands have extraneous output in the XML file. For example, the following is the extra output from `-get_cmos_dump`.

```
<SP_NETWORK_CONFIG>
  <DISCOVERY></DISCOVERY>
  <IP></IP>
  <NETMASK></NETMASK>
```

```
<GATEWAY></GATEWAY>
</SP_NETWORK_CONFIG>
<PASSWORD_CONFIG>
  <PASSWORD></PASSWORD>
</PASSWORD_CONFIG>
<BOOT_ORDER_OVERRIDE>
  <HELP_STRING>FIRST=Choose one of: pxe, cdrom, disk,
  floppy, bios, none</HELP_STRING>
  <FIRST></FIRST>
  <HELP_STRING>CLEAR_CMOS=Choose Yes, No or leave it
  empty, empty means No</HELP_STRING>
  <CLEAR_CMOS></CLEAR_CMOS>
</BOOT_ORDER_OVERRIDE>
<BOOT_DEVICE_PRIORITY>
  <B0>
    <DEVICE_NAME></DEVICE_NAME>
    <PCI-B-D-F></PCI-B-D-F>
  </B0>
</BOOT_DEVICE_PRIORITY>
```

5

Using fwupdate to Update Firmware

`fwupdate` is a utility that enables you to update, query, and validate the firmware of an Oracle server. This includes system firmware and the Oracle Integrated Lights Out Manager (ILOM), and device firmware such as network adapters, storage adapters, SAS expanders and various types of disk drives.

- [fwupdate Command Overview](#)
- [Listing Component Firmware Information](#)
- [Updating Component Firmware](#)
- [Reset a Device After a Firmware Update](#)
- [Execution Summary](#)

fwupdate Command Overview

This section covers the following information:

- [fwupdate Features](#)
- [fwupdate Command Prerequisites](#)
- [Secure Boot Setting Can Affect fwupdate Commands on Intel-Based Network and Storage Controllers](#)
- [Linux MMIO Access Settings Can Affect fwupdate Commands On Intel Network Controllers](#)
- [Downloading Firmware Patches](#)
- [fwupdate and Service Processor Access](#)
- [fwupdate Command Syntax](#)

fwupdate Features

`fwupdate` enables you to update firmware for the following components:

- System firmware and the Oracle ILOM service processor. System firmware includes BIOS for x86, and OBP, Hypervisor, NYX, POST, etc. for SPARC
- HBA and embedded storage controllers, SAS1, SAS2 and SAS3
- Disk drives (spinning media and flash drives)
- LSI SAS expander devices, SAS1, SAS2 and SAS3
- Emulex and QLogic Fiber Channel controllers

 **Note:**

Updating Emulex and QLogic Fiber Channel controller firmware using `fwupdate` requires the Emulex and QLogic vendor tools. If these packages are not already installed, install them as described in the *Oracle Hardware Management Pack Installation Guide*.

- Mellanox InfiniBand controllers
- Intel LOM (LAN on Motherboard)
- Oracle supported Network Interface Cards (NICs)

You can use `fwupdate` to do the following:

- List firmware information for devices in a server
- Ensure firmware file compatibility
- Update device firmware using an automated XML metadata file
- Update firmware manually using a raw firmware file
- Manually reset updated devices, if required

fwupdate Command Prerequisites

The following prerequisites must be met before using the `fwupdate` command:

- You must have root permission to run `fwupdate` commands on Unix-based platforms.
- On systems running Linux with the Secure Boot setting enabled in the BIOS Setup Utility, use the workaround listed in [Secure Boot Setting Can Affect fwupdate Commands on Intel-Based Network and Storage Controllers](#) before attempting to list or update Intel storage devices or controllers.
- On Linux systems with the kernel configured for strict memory mapped IO (MMIO) access (`iomem=strict`), use the workaround listed in [Linux MMIO Access Settings Can Affect fwupdate Commands On Intel Network Controllers](#) before attempting to list or update Intel network controllers.
- In order to update the firmware of InfiniBand adapter cards, such as the Oracle Dual Port QDR InfiniBand Adapter M3, using the Oracle Hardware Management Pack `fwupdate` tool, you must install the `mstflint` package. This package might be bundled with your operating system or available from Mellanox at: <https://www.mellanox.com/products/adapter-software/firmware-tools>.
- Before using the `fwupdate` command to update device firmware, you must quiesce the device.

 **Caution:**

System hang or data loss. Before updating device firmware, make sure that the device is quiesced.

- When updating the firmware on a hard drive, the following prerequisites must be met:
 - Make sure that the operating system is not accessing the disk (for example, the system boot disk).
 - Make sure that an application is not accessing the disk (for example, a database application).
 - If hardware RAID is being used on the system, make sure that the RAID controller is not accessing the disk (for example, if it is rebuilding an array or is in a degraded state). You can use `raidconfig` to check the state of the arrays.
 - For Oracle Solaris systems, after hot-plugging a device, run the `devfsadm -C` command to re-enumerate all of the system device nodes before running the `fwupdate` command.
- Updating Emulex and QLogic Fiber Channel controller firmware using `fwupdate` requires the Emulex and QLogic vendor tools provided with Oracle Hardware Management Pack. If they are not installed, you need to install them, refer to the *Oracle Hardware Management Pack 2.4 Installation Guide*.

Secure Boot Setting Can Affect fwupdate Commands on Intel-Based Network and Storage Controllers

On systems running Linux, when Secure Boot setting in the BIOS Setup Utility is enabled, `fwupdate` will not be able to see information on Intel-based storage devices and controllers. If `fwupdate` cannot detect information on a device, it cannot update its firmware. This affects Intel-based network controllers, front mounted NVMe SSDs and internal M.2 SSDs.

Note:

If you encounter this issue, you might also see an "MMIO memory settings" warning message even if you performed the workaround by setting kernel parameter "iomem=relaxed" as described in [Linux MMIO Access Settings Can Affect fwupdate Commands On Intel Network Controllers](#). In this case, you will also need to perform the Secure Boot workaround.

Workaround

Go into the BIOS Setup Utility and under the Security menu disable Secure Boot. Then restart the system, perform any firmware updates, and re-enable Secure Boot in the BIOS Setup Utility.

Linux MMIO Access Settings Can Affect fwupdate Commands On Intel Network Controllers

On some Linux kernels, memory mapped IO (MMIO) access is set to strict mode to prevent access to MMIO memory from user space utilities. When the kernel is running in this mode, `fwupdate` will not be able to access Intel-based network controllers.

If the `fwupdate` list or update command is run with kernel MMIO access is set to `iomem=strict`, the following can occur:

- fwupdate will display a warning to the console, for example:

```
# ./fwupdate list controller
```

```
WARNING: Due to strict MMIO memory settings in the running kernel
some network controllers may not be accessible.
See Hardware Management Pack documentation regarding iomem kernel
settings required for firmware update of these devices.
```

This warning informs you that fwupdate might not be able to access some network controllers. If fwupdate cannot access a device, it will not be able to list or update the device. As a result, no Intel network controllers will be listed in the fwupdate list controller output, though other updatable controllers will be listed.

- fwupdate produces a warning message in dmesg/syslog, for example:

```
kernel: Program fwupdate tried to access /dev/mem between c4a00000-
>c4a01000.
```

This is an informational message that fwupdate attempted to access restricted memory and will not cause an issue with the operation of the OS. There will be one of these messages logged each time fwupdate is run with the kernel in strict MMIO access mode.

- If the fwupdate update command is run in quiet (silent) mode, no messages are displayed to the console. A warning that the command failed can be captured in the output xml inventory when using the -o option. You can then see a <warning> entry for each device affected by this issue. For example, in the output xml file you might see:

```
<device>
  <devicetype>Controller</devicetype>
  <devicesubtype>NET</devicesubtype>
  <devicename>c10</devicename>
  <modelnumber>0x1528</modelnumber>
  <manufacturer>Intel</manufacturer>
  <node_id>Generic WWN:00:10:E0:57:7A:1E PCI Addr:82:00.0</node_id>
  <xml_support>N/A</xml_support>
  <pci_address>82:00.0</pci_address>
  <pci_vendor_id>0x8086</pci_vendor_id>
  <product_name>Intel X540 Gigabit Ethernet Controller</product_name>
  <warning>Firmware version data is not readable. Please see
Hardware Management Pack documentation and release notes for
more information on kernel configuration.</warning>
</device>
```

Workaround

If you encounter this issue, perform one of the following actions to allow fwupdate to access and list or update these devices.

- Add iomem=relaxed to the kernel command line.
- Run with a kernel version that does not restrict MMIO memory access to the user space.

Downloading Firmware Patches

Download firmware patches from <https://support.oracle.com>.

Search for the product that you want to update and download the latest firmware package available for that product.

fwupdate and Service Processor Access

When updating system firmware including the service processor (SP), `fwupdate` can be used over a local Host-to-ILOM interconnect or a remote Ethernet network connection as follows:

- When using local access, `fwupdate` uses the fastest local interface available. If a Host-to-ILOM connection is available this fast connection is used, otherwise the slower KCS interface is used. See [Host-to-ILOM Interconnect](#).

Note:

For systems with an Oracle ILOM version earlier than 3.2.4, you must manually include credentials using the `-H` and `-U` options (described below) for any commands that access a service processor. If credentials are not provided the commands will default to the slower local KCS interface to access the local service processor.

- When using remote Ethernet network access, `fwupdate` must present login credentials using a command line argument (SP hostname and user account with root access as described in [Command Options for Accessing Oracle ILOM Over a Remote Network Connection](#)). In addition, command execution over a remote network connection is encrypted using the TLS protocol. This means that a client-side trusted SSL certificate for the Oracle ILOM SP being accessed must be present on the host to validate the connection. This certificate checking feature is the default for a remote network connection when using the `fwupdate`, `ilomconfig` and `ubiosconfig` commands.
- [Obtaining SSL Certificates for TLS Access](#)
- [Command Options for Accessing Oracle ILOM Over a Remote Network Connection](#)

Obtaining SSL Certificates for TLS Access

In order to use TLS encryption when accessing a Oracle ILOM SP over a remote network connection, a client-side trusted certificate must be available on the host for the Oracle ILOM SP you will be accessing. Note the following:

- Ensure that you've installed the latest TLS and OpenSSL patches for your operating system (Oracle requires TLS 1.2 support at a minimum).
- Oracle Hardware Management Pack commands that perform SSL certificate validation for a remote network connection to a service processor look for hashed symbolic links to client-side trusted certificates in certain directories, depending on the operating system.
 - For Linux, the expected directory for certificate hashed links is `/etc/ssl/certs`.
 - For Oracle Solaris, the expected directory for certificate hashed links is `/etc/openssl/certs`.

If you opt to install certificates in some other directory, you will need to include a command line argument (as described in [Command Options for Accessing Oracle ILOM Over a Remote Network Connection](#)) that specifies the certificate directory when issuing

Oracle Hardware Management Pack commands that perform client-side SSL certificate validation.

To obtain a client-side trusted certificate from a service processor and prepare it for validation, do the following:

1. Obtain a PEM format certificate from the target Oracle ILOM SP. You can use one of the following methods:

- Use a web browser to log in to the Oracle ILOM SP. If this is the first time logging in to the SP, the browser will prompt you for a security exception at which point you can view and export the certificate in PEM format (.pem) to a directory.

If you've already accepted the certificate from a previous browser login, you can export it from the browser's stored servers certificates and export it in PEM format (.pem) to a directory.

- You can also run an OpenSSL command from the host to obtain the certificate. For example:

```
# echo | openssl s_client -connect sp_ip:623 | sed -n "/--BEGIN/,/--END/ p" > path_to_cert/certname.pem
```

Where *sp_ip* is the host name or IP address of the SP, *path_to_cert* is the directory path to where the certificate will be copied, and *certname* is the file name for the PEM format certificate.

Note:

To avoid the possibility of a man-in-the-middle attack, execute this command using a trusted channel or verified using an independent second channel.

- Or, set up your own certification authority and sign a certificate to upload to Oracle ILOM. If you choose to create your own custom certificates, refer to the Oracle ILOM documentation for details.
2. Change ownership of the certificate file you downloaded to `root:root` and file permissions to `-rw-r--r--` (numeric value 644).
 3. Create a hash symbolic link of your downloaded certificate.

- For Red Hat and Oracle Linux 6 and 7, there is a script called `cacertdir_rehash` which is part of the `authconfig` package. You can use this script to create your hash symbolic link in the certificate directory. For example:

```
# /usr/sbin/cacertdir_rehash /etc/ssl/certs
```

Refer to the Red Hat and Oracle Linux documentation for more details.

- For SLES Linux 12, there is a script called `c_rehash` which is part of the OpenSSL package. You can use this script to create your hash symbolic link in the certificate directory. For example:

```
# /usr/bin/c_rehash /etc/ssl/certs
```

Refer to the SLES Linux documentation for more details.

- For Oracle Solaris 11, rehash the certificate directory (this will also create the hash symbolic link in the `/etc/openssl/certs` directory). For example:

```
# /usr/sbin/svcadm refresh svc:/system/ca-certificates:default
```

Refer to the Oracle Solaris 11 documentation for more details.

- For Oracle Solaris 10, if you have installed the required patches the `/etc/openssl` directory should be present. Follow these steps to prepare the directory, hash the certificate and create the symbolic link:

- a. Create the `certs` directory under `/etc/openssl` if it does not already exist. Type:

```
# mkdir -p /etc/openssl/certs
```

- b. Set the group ownership for the directory. Type:

```
# chgrp sys /etc/openssl/certs
```

- c. Set permissions for the directory. Type:

```
# chmod 0755 /etc/openssl/certs
```

- d. Create the hash name for your certificate. Type:

```
# hashname=`openssl x509 -hash -noout -in path_to_cert/certname.pem`
```

Where *path_to_cert* is the directory path to your downloaded certificate and *certname* is the file name for the PEM format certificate.

- e. Create the symbolic link for your certificate. Type:

```
# ln -s "path_to_cert/certname.pem" "/etc/openssl/certs/$hashname.0"
```

Note:


The extension `.0` is an example, if there is another certificate using the `.0` extension, use the next available number for your extension.


Refer to the Oracle Solaris 10 and OpenSSL documentation for more details.

4. Ensure that the service processor Common Name (for example, ORACLESP-1000NML000) has been added to the domain name system (DNS) for your network. This name should match the Common Name found in the certificate file.

Command Options for Accessing Oracle ILOM Over a Remote Network Connection

The credential and certificate options listed in the following table are supported for `fwupdate` when accessing a service processor over a network connection. An example of usage follows the table.

Short Option	Long Options	Description
-H	--remote- hostname= <i>sp_ip</i>	<p>The host name, Common Name, or IP address of the remote service processor as specified by <i>sp_ip</i>. This option must be used in combination with the -U option.</p> <div data-bbox="1084 422 1380 1325" style="border: 1px solid #0070C0; padding: 10px; background-color: #E6F2FF;"> <p> Note:</p> <p>When accessing an SP over a remote Ethernet network connection, client-side SSL certificate validation is performed by default. For proper validation, you must use the Common Name stored in the client-side SSL certificate and the DNS server for the SP remote host name (e.g. -H ORACLESP-1000 NML000). Otherwise, you will receive a "hostname validation failed" error.</p> </div>
-U	--remote- username= <i>username</i>	The user name with root access used to log in to the remote service processor as specified by <i>username</i> . This option must be used in combination with the -H option.
-t	--intfname= <i>interface</i>	<p>Specifies the IPMI interface to use. No auto-detect is attempted. Supported interfaces that are compiled in are visible in the usage help output (socket interfaces in case -H option is used). See the -T description for more information.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.9.0.</i></p>

Short Option	Long Options	Description
-T	--remote-intfname-fallback= <i>interface</i>	<p>Selects the least secured IPMI socket interface to use if more secure interfaces are not supported. The tool attempts the most secure interface first (orcltIs). If the BMC does not support the interface, then attempt the next most secured socket interface until the specified interface. Supported socket interfaces that are compiled in are visible in the usage help output in the appropriate order. If lanplus or lan is specified, certificate checking is disabled when attempting the orcltIs interface.</p> <div data-bbox="1084 653 1378 1083" style="border: 1px solid #0070C0; padding: 10px; background-color: #E6F2FF;"> <p> Note:</p> <p>If the -T or -t option is not specified, then no auto-detect is enabled and only the orcltIs interface is attempted including certificate checking.</p> </div> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.9.0.</i></p>
n/a	--cert-dir= <i>pathname</i>	<p>Location of trusted certificates as specified by <i>pathname</i>. Use this option if your client-side SSL certificate is in a different directory than the expected default system certificate directory.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.4.</i></p>
n/a	--no-cert-check	<p>Do not perform SSL certificate checking.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.4.</i></p>

For example, where encryption is required for data transmitted over the network, use these command options to execute a command on a service processor over the network:

```
# fwupdate list controller --remote-hostname=sp_ip --remote-username=username --cert-dir=pathname
```

where *sp_ip* in this case is the Common Name for the target system's SP, *username* is the user name with login access rights to perform the operation, *pathname* is the path to the directory that contains your trusted certificate if it is not installed in the expected system certificate directory (see [Obtaining SSL Certificates for TLS Access](#)).

Once your certificate is validated and you are then prompted for the Oracle ILOM user password.

**Note:**

The Oracle ILOM user password required by the network connection can be piped in on stdin for scripting use.

fwupdate Command Syntax

The `fwupdate` command uses the following syntax:

```
fwupdate subcommand target options
```

If you use the `--help` or `--version` options, the `fwupdate` command does not require subcommands; otherwise a subcommand is mandatory.


When a command fails, it returns one of the failure codes listed in [fwupdate Error Codes](#).


`fwupdate` supports the subcommands listed in the following table.


Subcommand	Description
<code>list</code>	Provides firmware information about a device or a file. For targets and options specific to the <code>list</code> subcommand, see Listing Component Firmware Information .
<code>update</code>	Updates the firmware of one or more system components based on command-line directives. Devices can be updated automatically using metadata information contained in an XML file included with the device patch (recommended), or updated manually using a firmware image file. For targets and options specific to the <code>update</code> subcommand, see Updating Component Firmware .
<code>reset</code>	Resets the specified device if the device supports a reset. Perform a reset on a device after a manual mode firmware update (if required). In some cases where the metadata used in an automatic mode firmware update does not automatically reset a device that requires one, this subcommand can be used. For targets and options specific to the <code>reset</code> subcommand, see Reset a Device After a Firmware Update .

The following table lists general options available for `fwupdate`. Options specific to each subcommand are described in the section that describes using the subcommand.

Short Option	Long Option	Description
<code>-?, -h</code>	<code>--help</code>	Displays help information.

Short Option	Long Option	Description
-H	--remote- hostname= <i>sp_ip</i>	<p>The host name, Common Name, or IP address of the remote service processor as specified by <i>sp_ip</i>. This option must be used in combination with the -U option.</p> <div data-bbox="1084 422 1380 1327" style="border: 1px solid #0070C0; padding: 10px;"><p> Note:</p><p>When accessing an SP over a remote Ethernet network connection, client-side SSL certificate validation is performed by default. For proper validation, you must use the Common Name stored in the client-side SSL certificate and the DNS server for the SP remote host name (e.g. -H ORACLESP-1000 NML000). Otherwise, you will receive a "hostname validation failed" error.</p></div>
-U	--remote- username= <i>username</i>	<p>The user name with root access used to log in to the remote service processor as specified by <i>username</i>. This option must be used in combination with the -H option.</p>
-V	--version	<p>Displays the tool version.</p>

Short Option	Long Option	Description
-q	--quiet	Uses silent, non-interactive mode. Suppresses user prompts and informational message output and only returns error codes during the update. Useful for scripting. <div data-bbox="1084 422 1380 810"> Note: When using the quiet option, if the --no-cert-check option is not used and the certificate validation fails the utility will return an error.</div>
-t	--intfname= <i>interface</i>	Specifies the IPMI interface to use. No auto-detect is attempted. Supported interfaces that are compiled in are visible in the usage help output (socket interfaces in case -H option is used). See the -T description for more information. <i>This option was introduced in Oracle Hardware Management Pack 2.4.9.0.</i>

Short Option	Long Option	Description
-T	--remote-intfname-fallback= <i>interface</i>	<p>Selects the least secured IPMI socket interface to use if more secure interfaces are not supported. The tool attempts the most secure interface first (orcltIs). If the BMC does not support the interface, then attempt the next most secured socket interface until the specified interface. Supported socket interfaces that are compiled in are visible in the usage help output in the appropriate order. If lanplus or lan is specified, certificate checking is disabled when attempting the orcltIs interface.</p> <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> Note:</p> <p>If the -T or -t option is not specified, then no auto-detect is enabled and only the orcltIs interface is attempted including certificate checking.</p> </div> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.9.0.</i></p>
-y	--yes	Bypass user confirmation prompt when overwriting an existing output file of the same name.
n/a	--cert-dir= <i>pathname</i>	<p>Location of trusted certificates as specified by <i>pathname</i>. Use this option if your client-side SSL certificate is in a different directory than the expected default system certificate directory.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.4.</i></p>
n/a	--no-cert-check	<p>Do not perform SSL certificate checking.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.4.</i></p>

Listing Component Firmware Information

This section covers the following information:

- [list Subcommand Overview](#)
- [List All Component Firmware Information](#)
- [List Specific Component Firmware Information](#)

list Subcommand Overview

The `list` subcommand does the following:



- Displays the version of firmware for all components
- Tells you whether the target device can be updated with the XML metadata file
- Saves the configuration information to a specified XML file

This information can be used to check the state of a device before executing a firmware update and can be used to verify that a firmware update has been successful. The device naming convention used for target devices is shared with other Oracle Hardware Management CLI Tools.

The format for using the `list` subcommand is:

```
fwupdate list target options
```



The supported targets for the `list` subcommand are listed in the following table.

Target	Description
<code>all</code>	All supported component types, such as disks, expanders, controllers, bridge devices, and system firmware (including Oracle ILOM) that can be updated using an XML metadata file included with a firmware package.
<code>disk</code>	Supported hard disk drives and solid state disk drives.
<code>expander</code>	Supported SAS expanders.
<code>controller</code>	Supported controllers, such as storage and networking.  Note: Controllers that are not updatable by <code>fwupdate</code> will not be listed.
<code>bridge</code>	Supported embedded SAS-to-SATA bridge devices (used on some older systems).
<code>sp_bios</code>	System firmware on x86 or SPARC, including BIOS/OBP and Oracle ILOM.  Note: This target has been deprecated and is replaced by the <code>sysfw</code> target.

Target	Description
sp-bios-backup-firmware	Update the Oracle ILOM backup image and perform pre-update tasks as the first step to a BIOS/ILOM firmware update on an x86 system. The second step, loading the backup image and resetting the system to use the new firmware, is done from Oracle ILOM and can be delayed until a convenient time. <i>This option was introduced in Oracle Hardware Management Pack 2.4.5.</i>
sysfw	System firmware on x86 or SPARC, including BIOS/OBP and Oracle ILOM.
sysfw-backup-firmware	Update the Oracle ILOM backup image and perform pre-update tasks as the first step to a sysfw/ILOM firmware update on a SPARC system. The second step, loading the backup image and resetting the system to use the new firmware, is done from Oracle ILOM and can be delayed until a convenient time. <i>This option was introduced in Oracle Hardware Management Pack 2.4.5.</i>
fallback_boot	For SPARC systems with an SP that contains an updatable fallback boot image. <i>This target was introduced in Oracle Hardware Management Pack 2.4.</i>
supported-targets	List supported <code>fwupdate</code> component types that can be automatically updated using an XML metadata file. The firmware for these components can be updated individually or all at the same time using an XML metadata file included with a firmware package.
supported-images	List supported <code>fwupdate</code> component image types that can be manually updated. The firmware images of these components are individually updated by applying a firmware package image file.
error-codes	List all of the <code>fwupdate</code> return codes.

The `list` options are listed in the following table. When executing this command over a remote network connection, see [fwupdate and Service Processor Access](#).

Short Option	Long Option	Description
-n	--device_name	Allows a mandatory parameter to designate a single device to list. The <code>--device_name</code> option is the common-mapped device name. For more information, see CLI Tools Device-Naming Convention .
-v	--verbose	Displays detailed information about each component listed. Verbose is off by default.

Short Option	Long Option	Description
-x	--xml= <i>filename.xml</i>	<p>Uses the provided XML metadata file to determine which components are supported.</p> <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> Note:</p> <p>When the <code>-x</code> option is used to specify a metadata file, the devices discovered might change because the "software_type" specified in the metadata file may limit which devices are discovered. This might cause the device ID to change when specifying a metadata file with <code>-x</code> option.</p> </div>
-Q	--quick	<p>Optimizes access to include only the targeted devices during device discovery when using the <code>-x</code> option to specify a firmware metadata file. This can reduce the time it takes to execute the command.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.9.0.</i></p>
-o	-- output_xml= <i>filename.xml</i>	<p>Prints the configuration information in XML format to the given file.</p> <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> Note:</p> <p>This output option is not supported when using the list subcommand with the supported-targets, supported-images and error-codes targets. If you need to save information listed for these targets, you can redirect the standard display output to a file.</p> </div>

Short Option	Long Option	Description
-y	--yes	Bypass user confirmation prompt when overwriting an existing output file of the same name.

The following information is displayed with the `fwupdate list all` command. Items marked with an asterisk (*) are displayed in verbose listing.

- SP (SPARC) or SP + BIOS (x86)
 - ID
 - Product Name
 - ILOM Version
 - BIOS/OBP Version
 - Fallback Boot Version (SPARC systems that support it)
 - XML Support
- Controllers
 - ID
 - Type
 - Manufacturer
 - Mode
 - Product Name
 - Firmware (F/W) Version
 - BIOS version
 - EFI Version
 - FCODE Version
 - Package Version
 - NVDATA Version
 - XML Support
 - NODE ID*
 - Part Number*
 - PCI Address*
 - PCI Vendor ID*
 - WWN*
- Disk
 - ID
 - Manufacturer
 - Model
 - Chassis

- Slot
- Type
- Media
- Size
- Firmware (FW) Version
- XML Support
- NODE ID*
- WWN*
- Expander
 - ID
 - Chassis
 - Slot
 - Manufacturer
 - Model
 - Expander Name
 - Firmware (F/W) Version
 - XML Support
 - NODE ID*
 - Product Revision*
 - WWN*
- Bridge
 - ID
 - Chassis
 - Slot
 - Manufacturer
 - Model
 - Firmware (F/W) Version
 - Att FW Version
 - XML Support
 - NODE ID*
 - WWN*

List All Component Firmware Information

- To list all component firmware information on the system, type:

```
# fwupdate list all -v
```

The following is sample output from this command:

 **Note:**

The output for the c1 controller shows the NVMe controller type and the c2 and c3 controllers show NIC controller type, which are supported in Oracle Hardware Management Pack 2.3.1 and later.

```
=====
SP
=====
```

```
ID: sp
   Product Name: ORACLE SERVER X7-2L
   ILOM Version: v5.0.1.27 r139054
   BIOS/OBP Version: 42090300
   XML Support: N/A
```

```
=====
CONTROLLER
=====
```

```
ID: c0
   Node ID: mpt2sas:01:00.0
   Type: SAS
   Manufacturer: LSI Logic
   Model: 0x0072
   Product Name: SGX-SAS6-INT-Z
   FW Version: 11.05.02.00
   BIOS Version: 07.21.04.00
   EFI Version: 07.18.02.11
   PCI Address: 01:00.0
   PCI Vendor ID: 0x1000
   WWN: 0x500605b00452c5f0
   Serial Number: 500605b00452c5f0
   NVDATA Version: 10.03.00.26
   XML Support: N/A
   NAC Name: /SYS/MB/PCI2/SAS2
```

```
DISKS
=====
```

```
ID: c0d0
   Manufacturer: HGST
   Model: H101212SESUN1.2T
   Slot: 0
   Node ID: PDS:5000cca01d04e311
   Type: sas
   Media: HDD
   Size (GB): 1200
   Serial Number: 001304D2P9VD      KZG2P9VD
   FW Version: A447
   XML Support: N/A
   NAC Name: /SYS/HDD0
```

```
ID: c0d1
   Manufacturer: HGST
   Model: H101212SESUN1.2T
   Slot: 1
   Node ID: PDS:5000cca01d049199
   Type: sas
```

Media: HDD
Size (GB): 1200
Serial Number: 001304D2HWND KZG2HWND
FW Version: A447
XML Support: N/A
NAC Name: /SYS/HDD1

=====
CONTROLLER
=====

ID: c1
Node ID: nvme:81:00.00
Type: NVMe
Manufacturer: Intel
Model: 0x0953
Product Name: INTEL SSDPEDME016T4S
FW Version: 8DV1RA02
PCI Address: 81:00.0
PCI Vendor ID: 0x8086
Serial Number: CVMD4166002J1P6DGN
XML Support: N/A
NAC Name: /SYS/MB/PCI6/NVMe4

DISKS
=====

ID: c1d0
Manufacturer: INTEL
Model: SSDPEDME016T4S
Node ID: PDD:/dev/nvme0n1
Media: NVME
Size (GB): 200
Serial Number: CVMD4166002J1P6DGN
XML Support: N/A

=====
CONTROLLER
=====

ID: c2
Node ID: Generic WWN:00:10:E0:3B:F8:AC
Type: NET
Manufacturer: Intel
Model: 0x1528
Product Name: Intel(R) Ethernet Controller X540-AT2
EFI Version:
FCODE Version:
Package Version: 800004BE
PXE Version:
CLP Version:
FCOE Version:
ISCSI Version:
PCI Address: a0:00.0
PCI Vendor ID: 0x8086
Sequence Number: 0
XML Support: N/A
NAC Name: /SYS/MB/NET0

```
CONTROLLER
=====
ID: c3
Node ID: Generic WWN:00:10:E0:3B:F8:AE
Type: NET
Manufacturer: Intel
Model: 0x1528
Product Name: Intel(R) Ethernet Controller X540-AT2
EFI Version:
FCODE Version:
Package Version: 800004BF
PXE Version:
CLP Version:
FCOE Version:
ISCSI Version:
PCI Address: b0:00.0
PCI Vendor ID: 0x8086
Sequence Number: 1
XML Support: N/A
NAC Name: /SYS/MB/NET2
```

List Specific Component Firmware Information

- To list component firmware information, type:

```
fwupdate list target options
```

The following are some examples of the output for `fwupdate list` commands:

```
# fwupdate list disk -v

=====
CONTROLLER
=====
ID: c0
Type: HDC
Manufacturer: Intel
Model: 0xa182
Product Name: 0x486c
XML Support: N/A

DISKS
=====
ID: c0d0
Manufacturer: INTEL
Model: SSDSCKJB480G7
ATA Model: INTEL_SSDSCKJB480G7
Slot: 0
Type: sata
Media: SSD
Size(GiB): 480
Serial Number: 321222X          5XB1222X
FW Version: 0121
ATA FW Ver: N2010121
XML Support: N/A
NAC Name: /SYS/MB/RISER0/SSD0

=====
CONTROLLER
=====
```

```
ID: c1
  Node ID: mptir2:40:00.0
  Type: SAS
  Manufacturer: LSI Logic
  Model: 0x00ce
  Product Name: Avago MegaRAID SAS 9361-1
  FW Version: 11.05.02.00
  BIOS Version: 07.21.04.00
  EFI Version: 07.18.02.13
  FCODE Version: 01.00.60.00
  PCI Address: 40:00.0
  PCI Vendor ID: 0x1000
  WWN: 0x500605b005243000
  NVDATA Version: 10.03.00.26 (default) 10.03.00.27 (persistent)
  XML Support: N/A
  NAC Name: /SYS/MB/PCI2/SAS2
```

DISKS

=====

```
ID: c1d0
  Manufacturer: HITACHI
  Model: H106030SDSUN300G
  ATA Model: N/A
  Slot: 2
  Node ID: PDS:5000cca02515b089
  Type: sas
  Media: HDD
  Size (GB): 300
  FW Version: A2B0
  ATA FW Version: N/A
  XML Support: N/A
  NAC Name: /SYS/HDD0
```

```
ID: c1d1
  Manufacturer: HITACHI
  Model: H106030SDSUN300G
  ATA Model: N/A
  Slot: 3
  Node ID: PDS:5000cca025143f79
  Type: sas
  Media: HDD
  Size (GB): 300
  FW Version: A2B0
  ATA FW Version: N/A
  XML Support: N/A
  NAC Name: /SYS/HDD1
```

```
# fwupdate list sysfw -x metadata_5.0.1.27.xml
```

```
=====
SP
=====
ID          Product Name          System Firmware Version  ILOM
Version
  BIOSOBP Version  Fallback Boot Version  XML Support
-----
-
-----
sp          ORACLE SERVER X7-2L    -                          v5.0.1.27
```



```
r139054
  42090300          -          NA
```

```
# fwupdate list controller -n c1 -v
```

```
CONTROLLER
```

```
=====
```

```
ID: c1
  Node ID: mptmega:41:00.0
  Type: SAS
  Manufacturer: LSI Logic
  Model: 0x0079
  Product Name: LSI MegaRAID SAS 9261-8i
  FW Version: 2.130.353-1803
  BIOS Version: 3.24.00
  EFI Version: 4.12.05.00
  FCODE Version:
  PCI Address: 41:00.0
  PCI Vendor ID: 0x1000
  XML Support: N/A
  NAC Name: /SYS/MB/PCI2/SAS2
```

```
# fwupdate list disk -n c1d1
```

```
DISKS
```

```
=====
```

ID	Manufacturer	Model	ATA Model	Chassis Slot
Type	Media	Size(GiB)	FW Version	ATA FW Ver XML Support

c1d1	HITACHI	H7210A520SUN010T	-	-
N/A	sas	HDD	9124	A38K - NA

```
# fwupdate list disk -n c1d1 -v
```

```
DISK
```

```
=====
```

```
ID: c1d1
  Manufacturer: HITACHI
  Model: H7210A520SUN010T
  ATA Model: N/A
  Slot: 3
  Node ID: PDS:5000cca025143f79
  Type: sas
  Media: HDD
  Size (GB): 9124
  FW Version: A38K
  ATA FW Version: N/A
  XML Support: N/A
  NAC Name: /SYS/HDD1
```

```
# fwupdate list expander -n c1x0
```

```
EXPANDER
```

```
=====
```

ID	Chassis Slot	Manufacturer	Model	Expander Name	FW Version	XML Support
----	--------------	--------------	-------	---------------	------------	-------------

c1x0	0	-	ORACLE	DE2-24P	Primary	0010	N/A
------	---	---	--------	---------	---------	------	-----

```
# fwupdate list expander -n c1x0 -v
```

```
EXPANDER
=====
ID: clx0
  Chassis: 0
  Manufacturer: ORACLE
  Model: DE2-24P
  Expander Name: Primary
  FW Version: 0010
  Product Revision: 0010
  Node ID: EC:mpt2sas:30:00.0:5080020001431f3e
  XML Support: N/A
```

Updating Component Firmware

The following topics are covered in this section.

- [update Subcommand Overview](#)
- [Updating Component Firmware With a Metadata File \(Automatic Mode\)](#)
- [Updating Component Firmware Without a Metadata File \(Manual Mode\)](#)

update Subcommand Overview

The `update` subcommand is used with `fwupdate` to update component firmware. This might be system firmware (such as Oracle ILOM) and device firmware (such as a controller or disk drive). The `update` subcommand supports two methods for updating component firmware:

- **Automatic mode** uses information contained in an XML metadata file that is packaged with the patch to update the component firmware. This metadata contains information about the specific component(s) supported with the update and automates the update process to include any required host/device resets or power cycles. See the release notes included with the component firmware patch for more details. This is the most accurate and the recommended firmware update method to use.

Check the download package and its release notes to determine whether or not a metadata file is available, or simply look for the metadata file in the download package. The platform product notes might also contain important update information specific to the device you are updating.

- **Manual mode** allows you to update the firmware of a single component directly using a firmware file. This firmware update method should only be used if there is no XML metadata file available for the component that you want to update. In addition, any required device reset or power cycle not performed automatically after the update must be performed manually. Check the release notes included with the firmware for more details.

Each mode has its own supported command syntax and set of targets and options.

 **Note:**

Some systems require special processes to update system firmware. Always consult the release notes included with the firmware patch download for any special instructions that supersede instructions listed here.

Updating Component Firmware With a Metadata File (Automatic Mode)

`fwupdate update automatic` mode updates component firmware using information in a component-specific XML metadata file. Typically, each component firmware patch includes a metadata file as part of the patch download. The metadata method ensures that only components supported by the firmware patch get updated and also performs any device/host resets or host power cycles required. This is the most accurate method to use to update a firmware component in the system.

 **Note:**

Each component firmware patch includes its own component-specific metadata file. There are currently no metadata files that can be used to update all server components at once.


The following command syntax is used for the `update` subcommand when using automatic mode:

```
fwupdate update target options -x metadata.xml
```

Examples of command usage are listed below the supported target and options tables.


When used in automatic mode, the `update` subcommand supports the following component type targets:


Target	Description
<code>all</code>	All updatable component types specified in the XML metadata file. Typically, a firmware update patch for a single component, such as a controller, disk drive, or system SP, will include a component-specific metadata file. During the update process, all components of the same type and model described in the metadata file and found in the system will be updated. There are currently no metadata files that update all the firmware for all of the different components in a system at once.
<code>disk</code>	Supported hard disk drives or solid state disk drives.
<code>expander</code>	Supported SAS expanders.
<code>controller</code>	Supported controllers, such as storage and networking.
<code>bridge</code>	Supported embedded SAS-to-SATA bridge devices (used on some older systems).


Target	Description
sp_bios	System firmware on x86 (BIOS) or SPARC (OBP, Hypervisor, NYX, POST, etc.), including Oracle ILOM. <div style="border: 1px solid #0070c0; padding: 10px; background-color: #e6f2ff; margin: 10px 0;">  Note: This target has been deprecated and is replaced by the <code>sysfw</code> target. </div>
sysfw	System firmware on x86 (BIOS) or SPARC (OBP, Hypervisor, NYX, POST, etc.), including Oracle ILOM.
fallback_boot	For SPARC systems with an SP that contains an updatable fallback boot image. <i>This target was introduced in Oracle Hardware Management Pack 2.4.</i>

When used in automatic mode, `update` subcommand supports the options listed in the following table. When executing this command over a remote network connection, see [fwupdate and Service Processor Access](#).

Short Option	Long Option	Descriptions
-n	--device_name	Precedes name of the device to update. The name is the mapped name, which you can retrieve by using the <code>fwupdate list all</code> command. This option is not required when used with a metadata XML file. For information about device names, see CLI Tools Device-Naming Convention .
-d	--dry-run	Optional. Checks all input, executes an available dry-run check command on the firmware and component, but makes no permanent changes.

Short Option	Long Option	Descriptions
-x	--xml= <i>metadata.xml</i>	<p>If the firmware package contains a metadata XML file, this command provides the path to <i>metadata.xml</i>.</p> <div style="border: 1px solid #0070c0; padding: 10px; margin-top: 10px;"> <p> Note:</p> <p>When the -x option is used to specify a metadata file, the devices discovered might change because the "software_type" specified in the metadata file may limit which devices are discovered. This might cause the device ID to change when specifying a metadata file with -x option.</p> </div>
-Q	--quick	<p>Optimizes access to include only the targeted devices during device discovery when using the -x option to specify a firmware metadata file. This can reduce the time it takes to execute the command.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.9.0.</i></p>
-o	--output= <i>filename</i>	Logs all actions in the specified file.
-p	--end-priority= <i>value</i>	<p>Used with the <code>update</code> subcommand in automatic mode. End processing at a given priority level value in the metadata, skipping all levels with higher numeric values. For example, if you specify -p 3, only levels 1, 2 and 3 will be processed.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.</i></p>
-P	--start-priority= <i>value</i>	<p>Used with the <code>update</code> subcommand in automatic mode. Start processing at a given priority level value in the metadata, skipping all levels with lower numeric values. For example, if you specify -P 3, only levels 3 and higher (4, 5, etc.) will be processed.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.</i></p>

Short Option	Long Option	Descriptions
-q	--quiet	Uses silent, non-interactive mode. Suppresses user prompts and informational message output and only returns error codes during the update. Useful for scripting.
-y	--yes	Bypass user confirmation prompt when overwriting an existing output file of the same name. <i>This option was introduced in Oracle Hardware Management Pack 2.4.</i>
n/a	--silent-reboot	Enables a host reboot (or power cycle) after the firmware update with no prompt to the user. Reboot happens automatically. <div data-bbox="1084 688 1380 1386" style="border: 1px solid #0070C0; padding: 10px; background-color: #E6F2FF;"> <p> Note:</p> <p>This option is supported for all x86 systems. This option is also supported with newer SPARC systems that support Live Firmware Update and utilize metadata that includes a power cycle (refer to your SPARC firmware release notes to see if your system supports Live Firmware Update).</p> </div>

Short Option	Long Option	Descriptions
n/a	--silent-no-reboot	Prevents a host reboot after a firmware update. The user is not prompted to initiate a host reboot and no reboot takes place. If this option is used, the host will need to be manually rebooted later to complete the firmware update. <div data-bbox="1084 478 1377 1178" style="border: 1px solid #0070c0; padding: 10px; margin-top: 10px;"> <p> Note:</p> <p>This option is supported for all x86 systems. This option is also supported with newer SPARC systems that support Live Firmware Update and utilize metadata that includes a power cycle (refer to your SPARC firmware release notes to see if your system supports Live Firmware Update).</p> </div>
n/a	--fail-without-interconnect	Cancels a system firmware update if the Host-to-ILOM interconnect is not available. This prevents a default fallback to the using the much slower KCS interface for the update. <i>This option was introduced in Oracle Hardware Management Pack 2.4.1.</i>

Usage examples:

- To update the firmware of all devices supported in the metadata file, type:

```
# fwupdate update all -x metadata.xml
```

For example, if you are updating NVMe device firmware and there are three NVMe devices installed in the system, specifying `fwupdate update all` with an NVMe metadata file will update all three NVMe devices. This is the recommended and safest method for updating devices.

- To update system firmware (including Oracle ILOM) in quiet (non-interactive) mode but delay any host reboot or power cycle, type:

```
# fwupdate update sysfw -q -x metadata.xml --silent-no-reboot
```

This updates the platform system firmware as specified in the metadata without prompts or messages. If the metadata includes a host reboot or power cycle, the Oracle ILOM

service processor is restarted after the update, but the host system is not restarted. The system firmware update on the host (which includes updates to BIOS for x86, or OBP, Hypervisor, NYX, POST, etc. for SPARC) will be completed at the next power cycle for SPARC or reboot for x86.

 **Note:**

The `--silent-no-reboot` and `--silent-reboot` options are not supported on SPARC systems that do not support the Live Firmware Update feature. For systems that support Live Firmware Update, these options are supported if the metadata includes a power cycle. Check the release notes included with your SPARC firmware download package to see if this feature is supported.

- To update all disks supported in the metadata file, type:

```
# fwupdate update disk -x metadata.xml
```

This updates all disks in the system whose target device types are specified in the metadata file.

- To update a specific disk supported in the metadata file, type:

```
# fwupdate update disk -x metadata.xml -n c0d1
```

This updates only disk `c0d1`, and only if the `c0d1` disk drive type is specified in the metadata file.

See also

- [Update System Firmware Using Automatic Mode](#)
- [Update Device Firmware Using Automatic Mode](#)
- [Update a SPARC Fallback Image Using Automatic Mode](#)

Update System Firmware Using Automatic Mode

- To update system firmware using the fastest possible local option, ensure the Host-to-ILOM Interconnect is correctly configured to communicate with the target Oracle ILOM service processor ([Configuring the Host-to-ILOM Interconnect](#)). For systems that do not support updates over the Host-to-ILOM interconnect, such as the SPARC M series, the remote option utilizing a network connection to the service processor can be used.
- Download the firmware update from <https://support.oracle.com>.
The download should include a metadata file and firmware file for the target system.
- Make sure that the firmware is compatible with the target system. Some updates require a minimum version of firmware from which to perform the update. For some systems, special update processes might be described in the release notes that supersede instructions listed here. Read all documentation and release notes included with the firmware before proceeding.

A system firmware patch includes firmware for the Oracle ILOM service processor and either BIOS for x86 systems, or OBP, Hypervisor, NYX, POST, etc. for SPARC systems. In this context, system firmware refers to firmware required for base server

operation. It does not include firmware for ancillary devices such as controllers and disk drives.

1. To display information about system firmware, choose the local or remote option:

- *Local* – To list information about the local system firmware, type:

```
# fwupdate list sysfw
```

- *Remote* – To list information about system firmware using a network connection, type:

```
# fwupdate list sysfw -H sp_ip -U username
```

where *sp_ip* is the host name, Common Name (required for TLS encryption), or IP address of the service processor and *username* is the user name with Admin (a) role account privileges for logging in to Oracle ILOM.

Enter the Oracle ILOM password when prompted.

Output related to the target is displayed. For example the output from this command is similar to:

```
=====
SP
=====
ID          Product Name          System Firmware Version  ILOM Version
  BIOSOBP Version  Fallback Boot Version  XML Support
-----
sp          ORACLE SERVER X7-2L    -                          v5.0.1.27 r139054
42090300   -                          NA
```

2. To update the system firmware, choose the local or remote option:

- *Local* – To update the local system firmware, type one of the following commands:

```
- # fwupdate update all -x metadata.xml
```

--or--

```
- # fwupdate update sysfw -x metadata.xml
```

where *metadata.xml* is the path to the metadata file. To reduce the time it takes to perform the update, you can add the `-Q` option which optimizes access to include only the targeted devices during device discovery when specifying a firmware metadata file. For scripting purposes, you can add the `-q` option to perform the update without user interaction.

 **Note:**

For x86 systems, when updating system firmware you can add either the `--silent-reboot` or `--silent-no-reboot` option after the XML metadata file option to either automatically reboot or not automatically reboot the server after the firmware update.

For SPARC systems, these options can only be used if the system supports the Live Firmware Update feature and only if the metadata includes a power cycle. Check the release notes included with your SPARC firmware download package to see if this feature is supported.

- **Remote** – To update system firmware using a network connection, type one of the following commands:

```
– # fwupdate update all -x metadata.xml -H sp_ip -U  
  username  
  
  --or--  
  
– # fwupdate update sysfw -x metadata.xml -H sp_ip -U  
  username
```

where `metadata.xml` is the path to the metadata file, `sp_ip` is the host name, Common Name (required for TLS encryption), or IP address of the service processor and `username` is the user name with Admin (a) role account privileges for logging in to the Oracle ILOM service processor. To reduce the time it takes to perform the update, you can add the `-Q` option which optimizes access to include only the targeted devices during device discovery when specifying a firmware metadata file.

Enter the Oracle ILOM password when prompted.

 **Note:**

For x86 systems, when updating system firmware you can add either the `--silent-reboot` or `--silent-no-reboot` option after the XML metadata file option to either automatically reboot or not automatically reboot the server after the firmware update.

For SPARC systems, these options can only be used if the system supports the Live Firmware Update feature and only if the metadata includes a power cycle. Check the release notes included with your SPARC firmware download package to see if this feature is supported.

3. Follow any prompts, as required, to complete the update process.
4. If you opted not to automatically restart the server after the update, you must plan time to restart the server to utilize the new system firmware.

 **Note:**

Unless otherwise specified in the firmware release notes, a reboot is required for an x86 system; a power cycle is required for a SPARC system.

5. To confirm the new system firmware after the server has restarted, choose the local or remote option:
 - **Local** – To list information about the local system firmware, type:

```
# fwupdate list sysfw
```
 - **Remote** – To list information about system firmware using a network connection, type:

```
# fwupdate list sysfw -H sp_ip -U username
```

where *sp_ip* is the host name, Common Name (required for TLS encryption), or IP address of the service processor and *username* is the user name with Admin (a) role account privileges for logging in to the Oracle ILOM service processor.

Enter the Oracle ILOM password when prompted.

Output related to the target is displayed. For example the output from this command is similar to:

```
=====
SP
=====
ID          Product Name          System Firmware Version  ILOM Version
  BIOSOBP Version  Fallback Boot Version  XML Support
-----
sp          ORACLE SERVER X7-2L    -                          v5.0.1.27 r139054
  42090300      -                          NA
```

- [Update Device Firmware Using Automatic Mode](#)
- [Update a SPARC Fallback Image Using Automatic Mode](#)

Update Device Firmware Using Automatic Mode

- Download the firmware update from <https://support.oracle.com>.
The download should include a metadata file and firmware file for the target device.
- Make sure that the firmware is compatible with the target device; read all documentation and release notes included with the firmware before proceeding.
- Quiesce the device (stop all activity) before performing the update.

Device firmware includes storage and network controllers, disks, SAS expanders and SAS-to-SATA bridge devices.

1. To display information about component firmware, do one of the following:

- To list information about all components, type:
`# fwupdate list all`
- To list information about a specific device (such as a disk), type:
`# fwupdate list disk`

Output related to the target is displayed. For example the output from this command is similar to:

```
=====
CONTROLLER
=====
ID   Type   Manufacturer  Model   Product Name          FW Version
  BIOS Version  EFI Version   FCODE Version  Package Version  NVDATA Version
  XML Support
-----
-----
c0   SAS    LSI Logic     0x0072  SGX-SAS6-INT-Z        11.05.03.00
  07.21.09.00   07.22.05.00   01.00.62.00   -                10.03.00.28
  N/A
```

DISKS

```

=====
ID          Manufacturer  Model          ATA Model
Chassis Slot  Type   Media   Size(GiB)  FW Version ATA FW Ver XML Support
-----
-
-----
c0d0        SEAGATE          ST973401LSUN72G  -
-
      4      sas   HDD     68         0556      -         N/A
c0d1        SEAGATE          ST973401LSUN72G  -
-
      4      -    HDD     68         0556      -         N/A
c0d2        SEAGATE          ST973402SSUN72G  -
-
      5      sas   HDD     68         0603      -         N/A
c0d3        SEAGATE          ST973402SSUN72G  -
-
      5      -    HDD     68         0603      -         N/A

```

2. To update the device firmware, choose one of the following options:

- Update all devices supported in the metadata, type:

```
# fwupdate update all -x metadata.xml
```

where *metadata.xml* is the path to the metadata file. This updates all components in the system whose device types are specified in the metadata file.

- Update specific devices supported in the metadata, type:

```
# fwupdate update disk -x metadata.xml
```

where *metadata.xml* is the path to the metadata file. This updates all disks in the system whose device types are specified in the metadata file.

- Update a single device supported in the metadata, type:

```
# fwupdate update disk -x metadata.xml -n c0d1
```

where *metadata.xml* is the path to the metadata file. This updates only disk c0d1, and only if its disk drive type is specified in the metadata file.

3. Follow any prompts, as required, to complete the update process.

4. To confirm the new firmware, enter one of the following commands:

- To list information about all components, type:

```
# fwupdate list all
```

- To list information about a specific device, type:

```
# fwupdate list disk
```

- [Update System Firmware Using Automatic Mode](#)
- [Update a SPARC Fallback Image Using Automatic Mode](#)

Update a SPARC Fallback Image Using Automatic Mode

- To update the fallback image using the fastest possible local option, ensure the Host-to-ILOM Interconnect is correctly configured to communicate with the target Oracle ILOM service processor ([Configuring the Host-to-ILOM Interconnect](#)). For systems that do not support updates over the Host-to-ILOM interconnect, such as

the SPARC M series, the remote option utilizing a network connection to the service processor can be used.

- Download the firmware update from <https://support.oracle.com>.

The download should include a metadata file and firmware file for the target system.

- Make sure that the fallback image is compatible with the target system. Some updates require a minimum version of firmware from which to perform the update. For some systems, special update processes might be described in the release notes that supersede instructions listed here. Read all documentation and release notes included with the firmware before proceeding.

For some SPARC systems, the service processor contains an updatable fallback boot image. This image is stored in the SP and used only when the server is unable to locate its root device and boot media.

1. To display information about fallback boot image firmware, choose the local or remote option:

- *Local* – To list information about the local system fallback boot image firmware, type:

```
# fwupdate list fallback_boot -v
```

- *Remote* – To list information about fallback boot image firmware using a network connection, type:

```
# fwupdate list fallback_boot -v -H sp_ip -U username
```

where *sp_ip* is the host name, Common Name (required for TLS encryption), or IP address of the service processor and *username* is the user name with Admin (a) role account privileges for logging in to the Oracle ILOM service processor.

Enter the Oracle ILOM password when prompted.

Output related to the target is displayed. For example the output from this command is similar to:

```
=====
SP
=====
ID: sp
  Product Name: SPARC T7-1
  System Firmware Version: 9.5.2.g
  ILOM Version: v3.2.5.8.g r105871
  BIOS/OBP Version: OpenBoot 4.38.2 2015/10/30 13:09
  Fallback Boot Version: 11.3.10.5.0
  XML Support: N/A
```

2. To update the fallback boot image, choose the local or remote option:

- *Local* – To update the local fallback boot image firmware, type:

```
# fwupdate update fallback_boot -x metadata.xml
```

where *metadata.xml* is the path to the metadata file. For scripting purposes, you can add the *-q* option to perform the update without user interaction.

- *Remote* – To update fallback boot image firmware using a network connection, type:

```
# fwupdate update fallback_boot -x metadata.xml -H sp_ip -U
username
```

where *metadata.xml* is the path to the metadata file, *sp_ip* is the host name, Common Name (required for TLS encryption), or IP address of the service processor

and *username* is the user name with Admin (a) role account privileges for logging in to the Oracle ILOM service processor.

3. Follow any prompts, as required, to complete the update process.
4. To confirm the new fallback boot image firmware, choose the local or remote option:

- *Local* – To list information about the local system firmware, type:

```
# fwupdate list fallback_boot
```

- *Remote* – To list information about system firmware using a network connection, type:

```
# fwupdate list fallback_boot -H sp_ip -U username
```

where *sp_ip* is the host name, Common Name (required for TLS encryption), or IP address of the service processor and *username* is the user name with Admin (a) role account privileges for logging in to the Oracle ILOM service processor.

Enter the Oracle ILOM password when prompted.

Output related to the target is displayed. For example the output from this command is similar to:

```
=====
SP
=====
ID: sp
  Product Name: SPARC T7-1
  System Firmware Version: 9.5.2.g
  ILOM Version: v3.2.5.8.g r105871
  BIOS/OBP Version: OpenBoot 4.38.2 2015/10/30 13:09
  Fallback Boot Version: 11.3.11.6.0
  XML Support: N/A
```

- [Update System Firmware Using Automatic Mode](#)
- [Update Device Firmware Using Automatic Mode](#)

Updating Component Firmware Without a Metadata File (Manual Mode)

`fwupdate update manual mode` updates a single device using a firmware file that is part of the platform firmware download package. This method can be used if your firmware download package does not include a metadata file. Only one component and its supported firmware file can be specified per execution of this command.

Note:

You cannot update storage device firmware on the Flash Accelerator F40 PCIe Card directly. The Flash Accelerator F40 PCIe Card storage device firmware is updated when the general firmware package is applied to the PCIe card. An error is generated if you try to apply an update directly to the Flash Accelerator F40 PCIe Card storage devices.

The following command syntax is used for the update subcommand when using manual mode:


```
fwupdate update target options -n devicename -f filename
```

For manual mode, only one target device can be specified per `fwupdate` execution. Every target device that needs to be updated must be run as a separate `fwupdate` command.

Examples of command usage are listed below the supported target and options tables.

When used in manual mode, the `update` subcommand supports the following component type targets:

Target	Description
expander-firmware	Supported SAS expander.
expander-manufacturing-image	Supported SAS expander manufacturing image.
disk-firmware	Supported disk drive firmware.
sas-bridge-firmware	Supported SAS-to-SATA bridge device (used on some older systems).
sas-controller-firmware	Supported SAS controller firmware.
sas-controller-bios	Supported SAS controller BIOS code.
sas-controller-efi	Supported SAS controller EFI firmware.
sas-controller-fcode	Supported SAS controller fcode.
ib-controller-firmware	Supported InfiniBand adapter firmware.
fc-controller-firmware	Supported fibre channel adapter firmware.
nic-controller-firmware	Supported Ethernet network interface controller firmware. <i>This target was introduced in Hardware Management Pack 2.3.1.</i>


 **Note:**


This target is used to update add-in PCIe network adapters or onboard (also called LOM or LAN-on-motherboard) controller firmware. Currently, this feature is not available for all Intel-based Ethernet controllers or all operating systems. Refer to the Support Matrix for details at: <https://www.oracle.com/goto/ohmp>.

Target	Description
nvme-controller-firmware	Supported NVMe device firmware. <i>This target was introduced in Hardware Management Pack 2.3.1.</i>
sp-bios-firmware	System firmware on an x86 system, including BIOS and Oracle ILOM.
sp-bios-backup-firmware	Update the Oracle ILOM backup image and perform pre-update tasks as the first step to a BIOS/ILOM firmware update on an x86 system. The second step, loading the backup image and resetting the system to use the new firmware, is done from Oracle ILOM and can be delayed until a convenient time. <i>This option was introduced in Oracle Hardware Management Pack 2.4.5.</i>
sysfw-firmware	System firmware on a SPARC system, including OBP, Hypervisor, NYX, POST and Oracle ILOM. <i>This target was introduced in Hardware Management Pack 2.3.6.</i>
sysfw-backup-firmware	Update the Oracle ILOM backup image and perform pre-update tasks as the first step to a sysfw/ILOM firmware update on a SPARC system. The second step, loading the backup image and resetting the system to use the new firmware, is done from Oracle ILOM and can be delayed until a convenient time. <i>This option was introduced in Oracle Hardware Management Pack 2.4.5.</i>
fallback-boot-image	For SPARC systems that with an SP that contains an updatable fallback boot image. <i>This target was introduced in Oracle Hardware Management Pack 2.4.</i>

When used in manual mode, `update` subcommand supports the options listed in the following table. When executing this command over a remote network connection, see [fwupdate and Service Processor Access](#).

Short Option	Long Option	Descriptions
-n	--device_name	Name of the device to update. The name is the mapped name, which you can retrieve by using the <code>fwupdate list all</code> command. This option is mandatory for manual mode updates. For information about device names, see CLI Tools Device-Naming Convention .
-f	--filename=filename	A mandatory option, with a mandatory parameter, designating the name of the firmware image file that is to be applied.
-r	--reset	Reset the component after the update has finished.

Short Option	Long Option	Descriptions
-d	--dry-run	Optional. Checks all input, executes an available dry-run check command on the firmware and component, but makes no permanent changes.
-o	--output= <i>filename</i>	Log all actions in the specified file.
-q	--quiet	Uses silent, non-interactive mode. Suppresses user prompts and informational message output and only returns error codes during the update. Useful for scripting.
-y	--yes	Bypass any user confirmation prompt when overwriting an existing output file of the same name.
n/a	--fail-without-interconnect	Cancels a system firmware update if the Host-to-ILOM interconnect is not available. This prevents a default fallback to the using the much slower KCS interface for the update. <i>This option was introduced in Oracle Hardware Management Pack 2.4.1.</i>
n/a	--silent-reboot	Enables a host reboot (or power cycle) after the firmware update with no prompt to the user. Reboot happens automatically. <div data-bbox="1084 993 1378 1346" style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> Note: This option is supported for all x86 systems. This option is not supported for SPARC systems when using manual mode.</div>

Short Option	Long Option	Descriptions
n/a	--silent-no-reboot	Prevents a host reboot after a firmware update. The user is not prompted to initiate a host reboot and no reboot takes place. If this option is used, the host will need to be manually rebooted later to complete the firmware update. <div data-bbox="1084 478 1377 831" style="border: 1px solid #0070c0; padding: 10px; margin-top: 10px;"> <p> Note:</p> <p>This option is supported for all x86 systems. This option is not supported for SPARC systems when using manual mode.</p> </div>
n/a	--force	Ignore verification errors and proceed with updates anyway. Potentially dangerous, not recommended.

Usage examples:

- To update system firmware (including Oracle ILOM):

For a SPARC system, type: `fwupdate update sysfw-firmware -n devicename -f firmware-package-file.pkg`

For an x86 system, type: `fwupdate update sp-bios-firmware -n devicename -f firmware-package-file.pkg`

Where *devicename* is the device ID for the service processor (can be derived from the `fwupdate list all` command) and *firmware-package-file* is the name of the firmware file. This updates the platform system firmware, including Oracle ILOM. A host reboot (x86) or powercycle (SPARC) will need to be performed manually.

 **Note:**

For SPARC systems without the Live Firmware Update feature, the host power cycle (or power off) will be handled by Oracle ILOM.

- To update a specific disk using an image file, type:

```
# fwupdate update disk-firmware -n c1d1 -f diskfirmware.fw
```

This updates only disk `c1d1`.

See also

- [Update System Firmware Using Manual Mode](#)
- [Update Device Firmware Using Manual Mode](#)

Update System Firmware Using Manual Mode

- To update system firmware using the fastest possible local option, ensure the Host-to-ILOM Interconnect is correctly configured to communicate with the target Oracle ILOM service processor ([Configuring the Host-to-ILOM Interconnect](#)). For systems that do not support updates over the Host-to-ILOM interconnect, the remote option utilizing a network connection to the service processor can be used.
- Download the firmware update from <https://support.oracle.com>.
- Make sure that the firmware is compatible with the target system, read all documentation and release notes included with the firmware before proceeding. For some systems, special update processes might be described in the release notes that supersede instructions listed here.

System firmware is generally a bundled package that includes the base server firmware for the Oracle ILOM service processor and either BIOS for x86 systems, or OBP, Hypervisor, NYX, POST etc. for SPARC systems. Firmware for other components, such as controllers and disk drives are available as separate patches. Updating system firmware using manual mode can be used if your firmware download package does not include an XML metadata file.

1. To display information about system firmware, choose the local or remote option:

- *Local* – To list information about the local system firmware, type:

```
# fwupdate list sysfw
```

- *Remote* – To list information about system firmware using a network connection, type:

```
# fwupdate list sysfw -H sp_ip -U username
```

where *sp_ip* is the host name, Common Name (required for TLS encryption), or IP address of the service processor and *username* is the user name with Admin (a) role account privileges for logging in to the Oracle ILOM service processor.

Enter the Oracle ILOM password when prompted.

Output related to the target is displayed. For example the output from this command is similar to:

```
=====
SP
=====
ID          Product Name          System Firmware Version  ILOM Version
  BIOSOBP Version  Fallback Boot Version  XML Support
-----
sp          ORACLE SERVER X7-2L    -                          v5.0.1.27 r139054
42090300    -                          NA
```

2. To update the system firmware, choose the local or remote option:

- *Local* – To update the local system firmware, type one of the following commands:
 - For x86 systems: `fwupdate update sp-bios-firmware -n devicename -f fimware-package-file.pkg`
 - For SPARC systems: `fwupdate update sysfw-firmware -n devicename -f fimware-package-file.pkg`

Where *devicename* is the device ID for the service processor (derived in step 1) and *firmware-package-file* is the name of the firmware file. For scripting purposes, you can add the `-q` option to perform the update without user interaction.

This updates the platform system firmware, including Oracle ILOM. A host reboot (x86) or powercycle (SPARC) will need to be performed manually.

 **Note:**

For SPARC systems without the Live Firmware Update feature, the host power cycle (or power off) will be handled by Oracle ILOM.

- *Remote* – To update system firmware using a network connection, type one of the following commands:

```
For x86 systems: fwupdate update sp-bios-firmware -n
devicename -f firmware-package-file.pkg -H sp_ip -U
username
```

```
For SPARC systems: fwupdate update sysfw-firmware -n
devicename -f firmware-package-file.pkg -H sp_ip -U
username
```

Where *devicename* is the device ID for the service processor (derived in step 1) and *firmware-package-file* is the name of the firmware file, *sp_ip* is the host name, Common Name (required for TLS encryption), or IP address of the service processor and *username* is the user name with Admin (a) role account privileges for logging in to the Oracle ILOM service processor.

This updates the platform system firmware, including Oracle ILOM. A host reboot (x86) or powercycle (SPARC) will need to be performed manually.

 **Note:**

For SPARC systems without the Live Firmware Update feature, the host power cycle (or power off) will be handled by Oracle ILOM.

3. Follow any prompts, as required, to complete the update process.
4. Unless Oracle ILOM has performed a host power cycle (or power off) during the update process, restart the server to utilize the new system firmware after the update.

 **Note:**

Unless otherwise specified in the firmware release notes, a reboot is required for an x86 system; a power cycle is required for a SPARC system with the Live Firmware Update feature.

5. To confirm the new system firmware after the server has restarted, enter one of the following commands:

- *Local* – To list information about the local system firmware, type:

```
# fwupdate list sysfw
```

- *Remote* – To list information about system firmware using a network connection, type:

```
# fwupdate list sysfw -H sp_ip -U username
```

where *sp_ip* is the host name, Common Name (required for TLS encryption), or IP address of the service processor and *username* is the user name with Admin (a) role account privileges for logging in to the Oracle ILOM service processor.

Enter the Oracle ILOM password when prompted.

Output related to the target is displayed. For example the output from this command is similar to:

```
=====
SP
=====
ID          Product Name          System Firmware Version  ILOM Version
 BIOSOBP Version  Fallback Boot Version  XML Support
-----
sp          ORACLE SERVER X7-2L    -                          v5.0.1.27 r139054
42090300    -                          NA
```

Updating Component Firmware Without a Metadata File (Manual Mode)

Update Device Firmware Using Manual Mode

- Download the firmware update from <https://support.oracle.com>.
- Make sure that the firmware is compatible with the target device; read all documentation and release notes included with the firmware before proceeding.
- Quiesce the device (stop all activity) before performing the update.

Component firmware for specific devices, such as storage and network controllers, disks, SAS expanders and SAS-to-SATA bridge devices can be updated using manual mode if there is no XML metadata file included with the component firmware patch. Only one device and its supported firmware file can be specified per command.

1. To display information about component firmware, do one of the following:

- To list information about all components, type:

```
# fwupdate list all
```

- To list information about a specific device (such as a disk), type:

```
# fwupdate list disk
```

Output related to the target is displayed. For example the output from this command is similar to:

```
=====
CONTROLLER
=====
ID   Type   Manufacturer  Model   Product Name          FW Version
 BIOS Version  EFI Version   FCODE Version  Package Version  NVDATA Version
 XML Support
-----
```

```

-----
c0 SAS LSI Logic 0x0072 SGX-SAS6-INT-Z
11.05.03.00
07.21.09.00 07.22.05.00 01.00.62.00 -
10.03.00.28
N/A

DISKS
=====
ID Manufacturer Model ATA Model
Chassis Slot Type Media Size(GiB) FW Version ATA FW Ver XML Support
-----
-
-----
c0d0 SEAGATE ST973401LSUN72G -
-
4 sas HDD 68 0556 - N/A
c0d1 SEAGATE ST973401LSUN72G -
-
4 - HDD 68 0556 - N/A
c0d2 SEAGATE ST973402SSUN72G -
-
5 sas HDD 68 0603 - N/A
c0d3 SEAGATE ST973402SSUN72G -
-
5 - HDD 68 0603 - N/A

```

2. To update the device firmware, type:

```
# fwupdate update disk-firmware -n c0d1 -f diskfirmware.fw
```

Where *diskfirmware.fw* is the path to the firmware file. This updates only disk c0d1.

 **Note:**

If required, and the device supports a reset, add the `-r` option to reset the device after the update. Refer to the release notes for the firmware for details.

3. Follow any prompts, as required, to complete the update process.

4. Reset the device, if required, after the update.

Refer to the release notes that came with the firmware to see if a reset is necessary.

5. To confirm the new firmware, enter one of the following commands:

- To list information about all components, type:

```
# fwupdate list all
```

- To list information about a specific component, type:

```
# fwupdate list disk
```

- [Updating Component Firmware Without a Metadata File \(Manual Mode\)](#)
- [Reset a Device After a Firmware Update](#)

Reset a Device After a Firmware Update

After firmware for a device has been updated, the device might need to be reset. This requirement is different with each device; the reset functionality might be part of the update procedure or a separate function. To determine if your device requires a reset after a firmware update, consult the release notes included with your firmware.

- To reset a device, type:

```
fwupdate reset target -n devicename
```

The reset subcommand supports the following targets:

Target	Description
expander	Supported SAS expanders.
controller	Supported controllers, such as storage and networking.
sp_bios	The Oracle ILOM service processor on an x86 or SPARC system.
sysfw	The Oracle ILOM service processor on an x86 or SPARC system.

 **Note:**

This target has been deprecated and is replaced by the `sysfw` target.

Options for the `reset` subcommand are listed in the following table. When executing this command over a remote network connection, see [fwupdate and Service Processor Access](#).

Short Option	Long Option	Description
-n	--device_name	Required option followed by a mandatory parameter which designates a single device. Where the device name is the common-mapped device name shown when using the <code>fwupdate list all</code> command. For information about device names, see CLI Tools Device-Naming Convention .

Usage examples:

- To reset the Oracle ILOM service processor:

```
# fwupdate reset sysfw -n devicename
```

Where *devicename* is the device ID of the service processor as derived from the `fwupdate list all` command. This command only resets the service processor. It does not reset the host.

- To reset a specific controller, type:

```
# fwupdate reset controller -n c2
```

This command resets the controller identified as `c2` when using the `fwupdate list all` command.

Execution Summary

After the `fwupdate` tool is used to update firmware, an execution summary provides information on whether or not the update was successful. This information is also written to the log file.

The following examples show the possible execution summary messages:

- Message printed after a successful dry-run/check function:
`Check firmware successful for device: device_name`
- The update was successful, but no firmware version information is available for this component:
`Upgrade of firmware for device_name succeeded. Version information was not available.`
Consult your product release notes for information on how to verify the update.
- Update was successful:
`Upgrade of device_name from old_fw to new_fw succeeded.`
- The version number of the software did not change after a successful update:
`Upgrade of device_name from old_fw succeeded, but is not yet active.`
This might mean that the server needs to be reset, or that other instructions need to be followed. Consult your product release notes for instructions on how to update the version number.
- Update failed:
`Upgrade of device_name failed: error_message`

The variables in the previous output represent the following:

- `device_name` is the logical name of the device that is being updated.
- `old_fw` is the old firmware version.
- `new_fw` is the new firmware version.
- `error_message` is the error message that explains why the firmware update did not succeed.

6

Using hwmgmtcli to Display Hardware Information

`hwmgmtcli` displays hardware configuration information and the status of your Oracle servers.



Note:

There are some limitations to using `hwmgmtcli` tool for SPARC M5-32 and M6-32 servers. Refer to the *Release Notes* for more information.

- [hwmgmtcli Command Syntax](#)
- [List Subsystem Information](#)
- [View Open Problems](#)
- [Export Subsystem Information](#)

hwmgmtcli Command Syntax

The `hwmgmtcli` commands use the following command syntax:

```
hwmgmtcli subcommand subsystem [option]
```

The options listed in the following table apply to all CLI Tools commands, including `hwmgmtcli`.

Short Option	Long Option	Description
-?, -h	--help	Displays help information.
-V	--version	Displays the tool version.

If you use the `--help` or `--version` options, the `hwmgmtcli` command does not require subcommands, otherwise one or more subcommands are mandatory.

`hwmgmtcli` supports the subcommands shown in the following table.

Subcommand	Function
<code>list subsystem</code>	Show details of one or all subsystems.
<code>export all</code>	Export details of all subsystems to an XML file.

You can choose to show all available information or you can choose a subsystem. The available subsystems are listed in the following table.

Subsystem	Description
all	Show all subsystems available. For the <code>export</code> subcommand, this is the only supported subsystem.
server	Show details of server subsystem.
cooling	Show details of cooling subsystem.
processor	Show details of processor subsystem.
memory	Show details of memory subsystem.
power	Show details of power subsystem.
storage	Show details of storage subsystem.
network	Show details of network subsystem.
firmware	Show details of firmware subsystem.
device	Show details of the device subsystem.
bios	Show details of BIOS subsystem.
iomodule	Show details of IO module subsystem.
open_problems	Show all SP diagnosed open problems (ILOM 3.1 or newer).
dcu	Show details of dcu subsystem (only available on multi-domained systems).

The `list subsystem` subcommand supports the option listed in the following table.

Short Option	Long Option	Description
-d	--details	Show all of the properties and components for the subsystem in detail.

The option listed in the following table is supported for the `export all` subcommand.

Short Option	Long Option	Description
-f	--filename	Export the subsystem information to <code>filename.xml</code> .
-y	--yes	Bypass any user confirmation prompt when overwriting an existing output file of the same name.

List Subsystem Information

The `list` subcommand displays the current hardware configuration and status information of a server and its subsystems.

- To list subsystem information, type:


```
# hwmgmtcli list subsystem
```

 where *subsystem* is one of the subsystems listed in [hwmgmtcli Command Syntax](#).

The current subsystem information is listed.

View Open Problems

The `open_problems` subsystem displays information about logged system events.

- To view open server problems, type:

```
# hwmgmtcli list open_problems
```

The following display shows sample output from this command:

```
=== open_problems report ===
Open Problem 1
Problem time       : Thu Feb 14 22:38:19 2013
Problem subsystem  : System
Problem location   : /SYS (Host System)
Problem description : The top cover of server was opened while AC
input was still applied to the power supplies. (Probability: 100, UUID:
8bb87e70-d210-632b-d553-fc1450105bc4, Part Number: 31112054+1+1, Serial
Number: 1242FML0UV, Reference Document: http://support.oracle.com/msg/
SPX86-8003-8C).
Open Problem 2
Problem time       : Fri Feb 15 10:37:48 2013
Problem subsystem  : Storage
Problem location   : /SYS/DBP0/HDD2
Problem description : The disk temperature has exceeded the critical
limit. (Probability: 100, UUID: N/A, Part Number: H106030SDSUN300G, Serial
Number: 001234NTR1KD      PWGTR1KD, Reference Document: N/A)
```

Export Subsystem Information

The following procedure describes how to use the `export all` subcommand to save the current hardware configuration and status information of a server and its subsystems to a file.



Note:

The only subsystem available for the `export` subcommand is `all`.

- To export subsystem information, type:

```
# hwmgmtcli export all --filename filename.xml
```

where *filename* is the file to which you want to export the current system or subsystem information.

The current information is exported to the specified *filename.xml* file.

7

Using ilomconfig to Configure Oracle ILOM

`ilomconfig` allows you to configure Oracle ILOM service processors from the host OS without having to connect to the management network. You can target `ilomconfig` changes to either the local or a remote Oracle ILOM service processor.

`ilomconfig` also functions as an *XML builder* by either exporting the configuration of an Oracle ILOM service processor to an existing XML file, or creating a new XML file. These XML files can then be used for subsequent restore operations on compatible Oracle ILOM service processors.

You can also use `ilomconfig` to configure a Host-to-ILOM Interconnect on platforms that support this configuration. For more information on Host-to-ILOM Interconnect, see [Host-to-ILOM Interconnect](#).



Note:

There are some limitations to using `ilomconfig` tool for SPARC M5-32 and M6-32 servers. Refer to the *Release Notes* for more information.

- [ilomconfig Command Overview](#)
- [Importing and Exporting XML Configurations](#)
- [Listing System and SP Information](#)
- [Modifying Oracle ILOM Configurations](#)
- [Configuring the Host-to-ILOM Interconnect](#)
- [Managing Oracle ILOM and Host Watchdog](#)

ilomconfig Command Overview

This section covers the following information:

- [ilomconfig Features](#)
- [Restoring and Modifying Oracle ILOM XML Configuration Files](#)
- [ilomconfig and Service Processor Access](#)
- [ilomconfig Command Syntax](#)

ilomconfig Features

The `ilomconfig` commands can be directed at a local or remote Oracle ILOM service processor, or an XML configuration file. This file can then be used as a golden image to make changes to multiple Oracle ILOM service processors. You can either export the configuration of an Oracle ILOM service processor or create a new XML configuration file.

ilomconfig provides the following functions:

- Back up and restore from an Oracle ILOM XML file
- Modify the XML file using sub-commands
- Configure the network connection, including DHCP and sideband
- List and configure identification information, including hostname, contact, location, and description
- List and configure DNS
- List and configure clock including time zone
- List and configure user management
- List and configure SNMP community

Restoring and Modifying Oracle ILOM XML Configuration Files

Starting with Hardware Management Pack 2.1, ilomconfig can generate a backup of an Oracle ILOM service processor's configuration to an XML file with the `export config` command. The `create` or `modify` subcommands can be used to create or modify XML files.

By default, ilomconfig commands are executed on the local Oracle ILOM service processor. When you use the `--xmlfile=config.xml` option, the ilomconfig commands operate on the specified XML file.

The ilomconfig subcommands can modify already existing settings in the XML file or create new settings.

Note:

Ensure that when you create a new setting in an XML file, your target Oracle ILOM service processor supports the setting.

Oracle ILOM settings can be restored from an XML file starting with Oracle ILOM 3.0.12. Oracle ILOM settings that can be restored include:

- SSH private keys
- User SSH keys
- SSL cert
- COD license
- LDAP and AD certificates
- Platform binary data (currently limited to SPARC LDOMS config)
- User passwords
- SNMP users
- LDAP/LDAPSSL/RADIUS passwords
- Servicetag passphrase

ilomconfig and Service Processor Access

When accessing Oracle ILOM configurations on the service processor (SP), `ilomconfig` can be used over a local Host-to-ILOM interconnect or a remote Ethernet network connection as follows:

- When using local access, `ilomconfig` uses the fastest local interface available. If a Host-to-ILOM connection is available this fast connection is used, otherwise the slower KCS interface is used. See [Host-to-ILOM Interconnect](#).

Note:

For systems with an Oracle ILOM version earlier than 3.2.4, you must manually include credentials using the `-H` and `-U` options (described below) for any commands that access a service processor. If credentials are not provided the commands will default to the slower local KCS interface to access the local service processor.

- When using remote Ethernet network access, `ilomconfig` must present login credentials using a command line argument (SP hostname and user account with root access as described in [Command Options for Accessing Oracle ILOM Over a Remote Network Connection](#)). In addition, command execution over a remote network connection is encrypted using the TLS protocol. This means that a client-side trusted SSL certificate for the Oracle ILOM SP being accessed must be present on the host to validate the connection. This certificate checking feature is the default for a remote network connection when using the `fwupdate`, `ilomconfig` and `ubiosconfig` commands.
- [Obtaining SSL Certificates for TLS Access](#)
- [Command Options for Accessing Oracle ILOM Over a Remote Network Connection](#)

Obtaining SSL Certificates for TLS Access

In order to use TLS encryption when accessing a Oracle ILOM SP over a remote network connection, a client-side trusted certificate must be available on the host for the Oracle ILOM SP you will be accessing. Note the following:

- Ensure that you've installed the latest TLS and OpenSSL patches for your operating system (Oracle requires TLS 1.2 support at a minimum).
- Oracle Hardware Management Pack commands that perform SSL certificate validation for a remote network connection to a service processor look for hashed symbolic links to client-side trusted certificates in certain directories, depending on the operating system.
 - For Linux, the expected directory for certificate hashed links is `/etc/ssl/certs`.
 - For Oracle Solaris, the expected directory for certificate hashed links is `/etc/openssl/certs`.

If you opt to install certificates in some other directory, you will need to include a command line argument (as described in [Command Options for Accessing Oracle ILOM Over a Remote Network Connection](#)) that specifies the certificate directory when issuing Oracle Hardware Management Pack commands that perform client-side SSL certificate validation.

To obtain a client-side trusted certificate from a service processor and prepare it for validation, do the following:

1. Obtain a PEM format certificate from the target Oracle ILOM SP. You can use one of the following methods:

- Use a web browser to log in to the Oracle ILOM SP. If this is the first time logging in to the SP, the browser will prompt you for a security exception at which point you can view and export the certificate in PEM format (.pem) to a directory.

If you've already accepted the certificate from a previous browser login, you can export it from the browser's stored servers certificates and export it in PEM format (.pem) to a directory.

- You can also run an OpenSSL command from the host to obtain the certificate. For example:

```
# echo | openssl s_client -connect sp_ip:623 | sed -n "/--BEGIN/,/--END/ p" > path_to_cert/certname.pem
```

Where *sp_ip* is the host name or IP address of the SP, *path_to_cert* is the directory path to where the certificate will be copied, and *certname* is the file name for the PEM format certificate.

 **Note:**

To avoid the possibility of a man-in-the-middle attack, execute this command using a trusted channel or verified using an independent second channel.

- Or, set up your own certification authority and sign a certificate to upload to Oracle ILOM. If you choose to create your own custom certificates, refer to the Oracle ILOM documentation for details.
2. Change ownership of the certificate file you downloaded to `root:root` and file permissions to `-rw-r--r--` (numeric value 644).
 3. Create a hash symbolic link of your downloaded certificate.

- For Red Hat and Oracle Linux 6 and 7, there is a script called `cacertdir_rehash` which is part of the `authconfig` package. You can use this script to create your hash symbolic link in the certificate directory. For example:

```
# /usr/sbin/cacertdir_rehash /etc/ssl/certs
```

Refer to the Red Hat and Oracle Linux documentation for more details.

- For SLES Linux 12, there is a script called `c_rehash` which is part of the OpenSSL package. You can use this script to create your hash symbolic link in the certificate directory. For example:

```
# /usr/bin/c_rehash /etc/ssl/certs
```

Refer to the SLES Linux documentation for more details.

- For Oracle Solaris 11, rehash the certificate directory (this will also create the hash symbolic link in the `/etc/openssl/certs` directory). For example:

```
# /usr/sbin/svcadm refresh svc:/system/ca-certificates:default
```

Refer to the Oracle Solaris 11 documentation for more details.

- For Oracle Solaris 10, if you have installed the required patches the `/etc/openssl` directory should be present. Follow these steps to prepare the directory, hash the certificate and create the symbolic link:

- a. Create the `certs` directory under `/etc/openssl` if it does not already exist. Type:

```
# mkdir -p /etc/openssl/certs
```

- b. Set the group ownership for the directory. Type:

```
# chgrp sys /etc/openssl/certs
```

- c. Set permissions for the directory. Type:

```
# chmod 0755 /etc/openssl/certs
```

- d. Create the hash name for your certificate. Type:

```
# hashname=`openssl x509 -hash -noout -in path_to_cert/
certname.pem`
```

Where *path_to_cert* is the directory path to your downloaded certificate and *certname* is the file name for the PEM format certificate.

- e. Create the symbolic link for your certificate. Type:

```
# ln -s "path_to_cert/certname.pem" "/etc/openssl/
certs/$hashname.0"
```

 **Note:**


The extension `.0` is an example, if there is another certificate using the `.0` extension, use the next available number for your extension.


Refer to the Oracle Solaris 10 and OpenSSL documentation for more details.

4. Ensure that the service processor Common Name (for example, ORACLESP-1000NML000) has been added to the domain name system (DNS) for your network. This name should match the Common Name found in the certificate file.

Command Options for Accessing Oracle ILOM Over a Remote Network Connection

The credential and certificate options listed in the following table are supported for `ilomconfig` when accessing a service processor over a network connection. An example of usage follows the table.

Short Option	Long Options	Description
-H	--remote- hostname= <i>sp_ip</i>	<p>The host name, Common Name, or IP address of the remote service processor as specified by <i>sp_ip</i>. This option must be used in combination with the -U option.</p> <div data-bbox="1084 422 1382 1325" style="border: 1px solid #0070C0; padding: 10px; background-color: #E6F2FF;"> <p> Note:</p> <p>When accessing an SP over a remote Ethernet network connection, client-side SSL certificate validation is performed by default. For proper validation, you must use the Common Name stored in the client-side SSL certificate and the DNS server for the SP remote host name (e.g. -H ORACLESP-1000 NML000). Otherwise, you will receive a "hostname validation failed" error.</p> </div>
-U	--remote- username= <i>username</i>	The user name with root access used to log in to the remote service processor as specified by <i>username</i> . This option must be used in combination with the -H option.
-t	--intfname= <i>interface</i>	<p>Specifies the IPMI interface to use. No auto-detect is attempted. Supported interfaces that are compiled in are visible in the usage help output (socket interfaces in case -H option is used). See the -T description for more information.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.9.0.</i></p>

Short Option	Long Options	Description
-T	--remote-intfname-fallback= <i>interface</i>	<p>Selects the least secured IPMI socket interface to use if more secure interfaces are not supported. The tool attempts the most secure interface first (orcltls). If the BMC does not support the interface, then attempt the next most secured socket interface until the specified interface. Supported socket interfaces that are compiled in are visible in the usage help output in the appropriate order. If lanplus or lan is specified, certificate checking is disabled when attempting the orcltls interface.</p> <div data-bbox="1084 653 1378 1083" style="border: 1px solid #0070C0; padding: 10px; background-color: #E6F2FF;"> <p> Note:</p> <p>If the -T or -t option is not specified, then no auto-detect is enabled and only the orcltls interface is attempted including certificate checking.</p> </div> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.9.0.</i></p>
n/a	--cert-dir= <i>pathname</i>	<p>Location of trusted certificates as specified by <i>pathname</i>. Use this option if your client-side SSL certificate is in a different directory than the expected default system certificate directory.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.4.</i></p>
n/a	--no-cert-check	<p>Do not perform SSL certificate checking.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.4.</i></p>

For example, where encryption is required for data transmitted over the network, use these command options to execute a command on a service processor over the network:

```
# ilomconfig list system-summary --remote-hostname=sp_ip --remote-username=username --cert-dir=pathname
```

where *sp_ip* in this case is the Common Name for the target system's SP, *username* is the user name with login access rights to perform the operation, *pathname* is the path to the directory that contains your trusted certificate if it is not installed in the expected system certificate directory (see [Obtaining SSL Certificates for TLS Access](#)).

Once your certificate is validated and you are then prompted for the Oracle ILOM user password.

**Note:**

The Oracle ILOM user password required by the network connection can be piped in on stdin for scripting use.

ilomconfig Command Syntax

The `ilomconfig` commands must be run in administrator mode.


```
ilomconfig subcommand type [option]
```



When a command fails, it returns one of several failure codes listed in [ilomconfig Error Codes](#).

The available `ilomconfig` subcommands are listed in the following table.

Subcommand	Description
<code>list</code>	Show Oracle ILOM settings, users, SNMP communities, and system summary.
<code>create</code>	Create users and SNMP communities.
<code>delete</code>	Delete users and SNMP communities.
<code>modify</code>	Modify Oracle ILOM settings.
<code>import</code>	Restore Oracle ILOM settings from an XML file.
<code>export</code>	Backup Oracle ILOM settings to an XML file.
<code>reset</code>	Reset Oracle ILOM to factory defaults.
<code>enable</code>	Enable Host-to-ILOM interconnect. <div data-bbox="906 1220 1377 1465" data-label="Text"> <p> Note: The <code>enable/disable interconnect</code> subcommands cannot be used on remote service processors.</p> </div>
<code>disable</code>	Disable Host-to-ILOM interconnect. <div data-bbox="906 1577 1377 1822" data-label="Text"> <p> Note: The <code>enable/disable interconnect</code> subcommands cannot be used on remote service processors.</p> </div>

The following table lists general options available for `ilomconfig` and other Oracle Hardware Management Pack commands. Options specific to each subcommand are described in the section that describes using the subcommand.

Short Option	Long Option	Description
-?, -h	--help	Displays help information.
-H	--remote- hostname= <i>sp_ip</i>	The host name, Common Name, or IP address of the remote service processor as specified by <i>sp_ip</i> . This option must be used in combination with the -U option. <div data-bbox="1148 604 1198 646" style="float: right;">  </div> <div data-bbox="1198 604 1284 640" style="float: right;">Note:</div> <div data-bbox="1193 659 1438 1270" style="float: right; border: 1px solid #0070c0; padding: 5px; margin-top: 10px;"> <p>When accessing an SP over a remote Ethernet network connection, client-side SSL certificate validation is performed by default. For proper validation, you must use the Common Name stored in the client-side SSL certificate and the DNS server for the SP remote host name (e.g. -H ORACLESP-1000NML000). Otherwise, you will receive a "hostname validation failed" error.</p> </div>
-U	--remote- username= <i>username</i>	The user name with root access used to log in to the remote service processor as specified by <i>username</i> . This option must be used in combination with the -H option.
-V	--version	Displays the tool version.

Short Option	Long Option	Description
-q	--quiet	<p>Uses silent, non-interactive mode. Suppresses user prompts and informational message output and only returns error codes.</p> <div data-bbox="1117 394 1461 751" style="border: 1px solid #0070c0; padding: 10px; background-color: #e6f2ff;"> <p> Note:</p> <p>When using the quiet option, if the --no-cert-check option is not used and the certificate validation fails the utility will return an error.</p> </div>
-t	--intfname= <i>interface</i>	<p>Specifies the IPMI interface to use. No auto-detect is attempted. Supported interfaces that are compiled in are visible in the usage help output (socket interfaces in case -H option is used). See the -T description for more information.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.9.0.</i></p>
-T	--remote-intfname-fallback= <i>interface</i>	<p>Selects the least secured IPMI socket interface to use if more secure interfaces are not supported. The tool attempts the most secure interface first (orcltIs). If the BMC does not support the interface, then attempt the next most secured socket interface until the specified interface. Supported socket interfaces that are compiled in are visible in the usage help output in the appropriate order. If lanplus or lan is specified, certificate checking is disabled when attempting the orcltIs interface.</p> <div data-bbox="1117 1346 1461 1665" style="border: 1px solid #0070c0; padding: 10px; background-color: #e6f2ff;"> <p> Note:</p> <p>If the -T or -t option is not specified, then no auto-detect is enabled and only the orcltIs interface is attempted including certificate checking.</p> </div> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.9.0.</i></p>
-y	--yes	Execute command without prompting for confirmation.

Short Option	Long Option	Description
n/a	<code>--cert-dir=pathname</code>	Location of trusted certificates as specified by <i>pathname</i> . Use this option if your client-side SSL certificate is in a different directory than the expected default system certificate directory. <i>This option was introduced in Oracle Hardware Management Pack 2.4.4.</i>
n/a	<code>--no-cert-check</code>	Do not perform SSL certificate checking. <i>This option was introduced in Oracle Hardware Management Pack 2.4.4.</i>

Importing and Exporting XML Configurations

This section covers the following information:

- [Export an XML Configuration](#)
- [Import an XML Configuration](#)

Export an XML Configuration

To export an entire configuration to an XML file, use the `ilomconfig export config` command.

Note:

Exit or close all active ILOM login sessions before proceeding. The `ilomconfig export file` command enables exports of the current Oracle ILOM configuration. Before an export operation can be executed, all active open sessions must be closed. There must be no active ILOM sessions logged in to `/SP/console` during the export operation.

Note:

If the `--xmfile` option is specified, the changes are only made to the XML file. If the XML file option is omitted, the changes are made directly to the Oracle ILOM. The XML file cannot be manually edited by a user, it can only be changed by using `ilomconfig`.

- Do one of the following:

 **Note:**

To back up sensitive data such as passwords, SSH keys, certificates, LDoms and so forth, you must specify a passphrase. **The passphrase length must be a minimum of 16 characters.**

- To export an XML configuration using a passphrase, choose one of the following commands:
 - To get a prompt asking whether you want to enter a passphrase:

```
# ilomconfig export config --xmlfile=filename.xml
```

where *filename* represents the file to which you are exporting the ILOM configuration.
For example:

```
# ilomconfig export config --xmlfile=config.xml
Do you want to enter a passphrase to back up sensitive data? [y/n]?
Y
Enter passphrase: *****
Wrote backup of ILOM configuration to 'config.xml'.
```
 - To set up a passphrase to be used with an automated script, provide a passphrase or a file containing the passphrase as follows:

```
# echo passphrase | ilomconfig export config --
xmlfile=filename.xml
```

where *passphrase* is the passphrase that you want to use.
or

```
# cat file_with_passphrase | ilomconfig export config
--xmlfile=filename.xml
```

where *file_with_passphrase* is the file containing the passphrase.
For example:

```
# echo passphrase | ilomconfig export config --
xmlfile=config.xml
Enter passphrase: *****
Wrote backup of ILOM configuration to 'config.xml'.
```

The passphrase is automatically passed through the command line.
- To export an XML configuration without using a passphrase:

```
# ilomconfig export config --xmlfile=filename.xml -y
```

where *filename* represents the file to which you are exporting the ILOM configuration.
For example:

```
# ilomconfig export config --xmlfile=config.xml -y
Wrote backup of ILOM configuration to 'config.xml'.
```

This option exports the ILOM configuration without using a passphrase.

Import an XML Configuration

To import an XML configuration file to configure Oracle ILOM, use the `ilomconfig import config` command. You can also use this command to restore the system configuration by importing a known reliable XML file.

Note:

Exit or close all active ILOM login sessions before proceeding. The `ilomconfig import file` command imports the current Oracle ILOM configuration. Before an import operation can be executed, all active open sessions must be closed. There must be no active ILOM sessions logged in to `/SP/console` during the import operation.

Note:

If the `--xmlfile` option is specified, the changes are only made to the XML file. If the XML file option is omitted, the changes are made directly to the Oracle ILOM. The XML file cannot be manually edited by a user, it can only be changed by using `ilomconfig`.

- Do one of the following:
 - To import an XML configuration using a passphrase, choose one of the following commands.

- To get a prompt asking whether you want to enter a passphrase, type:

```
# ilomconfig import config --xmlfile=filename.xml
```

where *filename* represents the file from which you are importing the ILOM configuration.

For example:

```
# ilomconfig import config --xmlfile=config.xml
Are you sure you want to import the settings from the XML file to ILOM?
[y/n]? y
Do you want to enter a passphrase to restore sensitive data? [y/n]? y
Enter passphrase: *****
Preparing to restore XML file to ILOM...
Done preparing to restore XML file ILOM.
Restoring configuration (allow several
minutes).....
.....
..
.....Done.
```

- To set up a passphrase to be used with an automated script, provide a passphrase or a file containing the passphrase as follows:

```
# echo passphrase | ilomconfig import config --
xmlfile=filename.xml
```


where *passphrase* is passphrase that you want to use.

or

```
# cat file_with_passphrase | ilomconfig import config
--xmlfile=filename.xml
```

where *file_with_passphrase* is the file containing the passphrase.

For example:

```
# echo passphrase | ilomconfig import config --
xmlfile=config.xml
Enter passphrase: *****
Preparing to restore XML file to ILOM...
Done preparing to restore XML file ILOM.
Restoring configuration (allow several
minutes).....
.....
.....Done.
```

The passphrase is automatically passed in through the command line.

- To import an XML configuration without using a passphrase, type:

```
# ilomconfig import config --xmlfile=filename.xml -y
```

where *filename* represents the file from which you are importing the ILOM configuration.

For example:

```
# ilomconfig import config --xmlfile=config.xml -y
Preparing to restore XML file to ILOM...
Done preparing to restore XML file ILOM.
Restoring configuration (allow several
minutes).....
.....
.....Done.
```

This option imports the ILOM configuration without using a passphrase.

Listing System and SP Information

This section covers the following information:

Note:

If the `--xmfile` option is specified, the command is executed on the XML file. If the XML file option is omitted, the command is executed on Oracle ILOM. The XML file cannot be manually edited by a user, it can only be changed by using `ilomconfig`.

- [List System Summary Information](#)
- [List Users](#)

- [List an SNMP Community](#)
- [List IPv4 Network Settings](#)
- [List IPv6 Network Settings](#)
- [List Service Processor Identification Information](#)
- [List DNS Information](#)
- [List Clock Information](#)

List System Summary Information

Use the `ilomconfig list` sub command to list system summary information including the product name, part number, serial number, Oracle ILOM host name, and Oracle ILOM version information. Use the `ilomconfig list system-summary` command to lists the same information as the Summary tab in the Oracle ILOM web interface.

- Choose one of these procedures depending on where the system summary details are:

- To view the system summary of the local Oracle ILOM service processor, type:

```
# ilomconfig list system-summary
```

- To view the system summary information from a remote Oracle ILOM service processor, type:

```
# ilomconfig list system-summary --remote-hostname=sp_ip --  
remote-username=username
```

where *sp_ip* is the host name, Common Name (required for TLS encryption), or IP address of the remote server's service processor and *username* the valid user account with privileges to view system summary information.

For example:

```
# ilomconfig list system-summary --remote-hostname=192.0.2.10  
--remote-username=root
```

Oracle ILOM prompts for the root account password.

List Users

To list one or all users, use the `ilomconfig list user username` command. If *username* is specified then only that user is listed. If *username* is blank, then all users are listed.

When you specify an XML file name, the command is run on information available in the exported service processor configuration XML file rather than querying Oracle ILOM.

- To list users, type:

```
# ilomconfig list user [username] [--xmlfile=filename.xml]
```

where *username* is the user to list and *filename* is the name of the service processor configuration XML file.

List an SNMP Community

To list one or all SNMP communities, use the `ilomconfig snmp-community` command. When you specify an XML file name, the command lists SNMP communities defined in the exported service processor configuration XML file rather than querying Oracle ILOM.

- To list SNMP communities, type:

```
# ilomconfig list snmp-community [communityname] [--  
xmlfile=filename.xml]
```

where *communityname* is the name of the SNMP community you are interested in and *filename* is the name of the service processor configuration XML file.

List IPv4 Network Settings

To list IPv4 network settings, use the `ilomconfig list network` command. This command lists IP address, netmask, gateway, DHCP settings, sideband, and MAC. When you specify an XML file name, this command lists IPv4 network settings defined in the exported service processor configuration XML file rather than querying Oracle ILOM.

- To list IPv4 network settings, type:

```
# ilomconfig list network [--xmlfile=filename.xml]
```

List IPv6 Network Settings

To list IPv6 network settings, use the `ilomconfig list network-ipv6` command. This command lists IP address, gateway, autoconfig, link local IP address, dynamic IP address and interface state. When you specify an XML file name, this command lists IPv6 network settings defined in the exported service processor configuration XML file rather than querying Oracle ILOM.

- To list IPv6 network settings, type:

```
# ilomconfig list network-ipv6 [--xmlfile=filename.xml]
```

List Service Processor Identification Information

To list identification information for the service processor, use the `ilomconfig list identification` command. This command lists service processor host name, system contact, system location, and system description, which is equivalent to the Identification tab on web interface. When you specify an XML file name, the command lists identification information defined in the exported service processor configuration XML file rather than querying Oracle ILOM.

- To list service processor identification information, type:

```
# ilomconfig list identification [--xmlfile=filename.xml]
```

List DNS Information

To list DNS information, use the `ilomconfig list dns` command. If you specify an XML file name, the command lists DNS information defined in the exported service processor configuration XML file rather than querying Oracle ILOM itself.

- To list DNS information, type:

```
# ilomconfig list dns [--xmlfile=filename.xml]
```

List Clock Information

To list clock information, use the `ilomconfig list clock` command. When you specify an XML file name, the command lists clock information defined in the exported service processor configuration XML file rather than querying Oracle ILOM.

- To list clock information, type:

```
# ilomconfig list clock [--xmlfile=filename.xml]
```

Modifying Oracle ILOM Configurations

This section covers the following information:



Note:

If the `--xmfile` option is specified, the changes are only made to the XML file. If the XML file option is omitted, the changes are made directly to the Oracle ILOM. The XML file cannot be manually edited by a user, it can only be changed by using `ilomconfig`.

- [Restore Oracle ILOM to Defaults](#)
- [Create a User](#)
- [Delete a User](#)
- [Modify a User Password or Role](#)
- [Create an SNMP Community](#)
- [Modify IPv4 Network Settings](#)
- [Modify IPv6 Network Settings](#)
- [Modify Identification Information](#)
- [Modify DNS Information](#)
- [Modify Clock Information](#)

Restore Oracle ILOM to Defaults

To restore the Oracle ILOM configuration to the factory defaults, use the `ilomconfig reset config` command. Use the `-y` option to bypass the yes or no confirmation prompt. This results in the reboot of the Oracle ILOM.

- To restore Oracle ILOM to defaults, type:

```
# ilomconfig reset config [-y]
```

Create a User

To create a user, use the `ilomconfig create user` command. The `-y` option prevents the yes/no confirmation prompt. When you specify an XML file name, the command modifies information defined in the exported service processor configuration XML file rather than modifying Oracle ILOM.

1. To create a user, type:

```
# ilomconfig create user username [-y][--role=role] [--xmlfile=filename.xml]
```

where *username* is the user to modify, *role* is the role of the Oracle ILOM user and *filename* is the name of the exported service processor configuration XML file to modify.

2. At the prompt, enter the password for the user.

Delete a User

To delete a user, use the `ilomconfig delete user` command. The `-y` option prevents the yes or no confirmation prompt. When you specify an XML file name, the command modifies information defined in the exported service processor configuration XML file rather than modifying Oracle ILOM.

- To delete a user, type:

```
# ilomconfig delete user username [-y] [--xmlfile=filename.xml]
```

where *username* is the user to delete and *filename* is the name of the exported service processor configuration XML file to modify.

Modify a User Password or Role

To modify a user password or role, use the `ilomconfig modify user` command. When you specify an XML file name, the command modifies information defined in the exported service processor configuration XML file rather than modifying Oracle ILOM.

- To modify a user password or role, type:

```
# ilomconfig modify user username [-p] [--role=role] [--xmlfile=filename.xml]
```

where *username* is the user to modify, *-p* prompts for the user's password, *role* is the role of the Oracle ILOM user and *filename* is the name of the exported service processor configuration XML file to modify.

Create an SNMP Community

To create an SNMP community, use the `ilomconfig create snmp-community` command. When you specify an XML file name, the command modifies information defined in the exported service processor configuration XML file rather than modifying Oracle ILOM.

- To create an SNMP community, type:

```
# ilomconfig create snmp-community communityname [--permission=ro|rw] [--xmlfile=filename.xml]
```

where *communityname* is the SNMP community you are creating, *--permission* is either read-only or read-write (*ro|rw*), and *filename* is the name of the exported service processor configuration XML file to modify.

Note:

Starting with Oracle ILOM 4.0, `ilomconfig` will no longer be able to create SNMP communities with read/write (*rw*) permissions. Only the read-only (*ro*) permission is allowed.

Modify IPv4 Network Settings

To modify IPv4 settings, use the `ilomconfig modify network` command. This command modifies IP address, netmask, gateway, DHCP settings, and sideband. When you specify an XML file name, the command modifies information defined in the exported service processor configuration XML file rather than modifying Oracle ILOM.

- To modify IPv4 network settings, type:

```
# ilomconfig modify network [--ipdiscovery=static|dhcp] [--ipaddress=ipaddress] [--netmask=netmask] [--gateway=gateway] [--state=enabled|disabled] [--mgmtport=port] [--xmlfile=filename.xml]
```

Option	Description	Example
<code>--ipdiscovery</code>	Network discovery mechanism. Can be either static or DHCP.	<code>static</code> or <code>dhcp</code>
<code>--ipaddress</code>	Oracle ILOM IP address	<code>192.0.2.10</code>
<code>--netmask</code>	Netmask address	<code>255.255.255.0</code>
<code>--gateway</code>	Gateway address	<code>192.0.2.248</code>
<code>--state</code>	Oracle ILOM management port state	<code>enabled</code> or <code>disabled</code>
<code>--mgmtport</code>	Oracle ILOM management port path	<code>/SYS/SP/NET0</code> or <code>SYS/MB/SP/NETMGMT</code>

Option	Description	Example
<code>--xmlfile</code>	Modify specified XML file rather than local Oracle ILOM service processor. Must be followed by = and the pathname to the file.	<i>file.xml</i>

Modify IPv6 Network Settings

To modify IPv6 settings, use the `ilomconfig modify network-ipv6` command. This command lists IP address, netmask, gateway, DHCP settings, and sideband. When you specify an XML file name, the command modifies information defined in the exported service processor configuration XML file rather than modifying Oracle ILOM.

- To modify IPv6 network settings, type:

```
# ilomconfig modify network-ipv6 [--static-  
ipaddress=IPv6_address] [--autoconfig=disabled|stateless|  
dhcpv6_stateful|dhcpv6_stateless] [--state=enabled|disabled]  
[--xmlfile=filename.xml]
```

Option	Description	Example
<code>--static-ipaddress</code>	Oracle ILOM IPv6 static address.	2001:0db0:0000:82a1: :0000:0000:1234:abcd
<code>--autoconfig</code>	Oracle ILOM IPv6 autoconfiguration state.	When using Oracle ILOM 3.0.12.x: disabled, stateless_only When using Oracle ILOM 3.0.14.x: disabled, stateless, dhcpv6_stateful, dhcpv6_stateless
<code>--state</code>	Oracle ILOM IPv6 administrative state.	enabled or disabled
<code>--xmlfile</code>	Modify specified XML file rather than local Oracle ILOM service processor. Must be followed by = and the pathname to the file.	<i>file.xml</i>

Modify Identification Information

To modify identification information, use the `ilomconfig modify identification` command. This command modifies the host name, system contact, system location, and system description. When you specify an XML file name, the command modifies information defined in the exported service processor configuration XML file rather than modifying Oracle ILOM.

- To modify identification information, type:

```
# ilomconfig modify identification [--hostname=hostname] [--system-contact=system_contact] [-system-location=system_location] [--system-identifier=system_identifier] [--xmlfile=filename.xml]
```

Option	Description	Example
--hostname	Oracle ILOM host name.	service-processor.domain.com
--system-contact	Oracle ILOM system contact field.	user
--system-location	Oracle ILOM system location field.	west
--system-identifier	Oracle ILOM system identifier field.	x4800
--xmlfile	Modify specified XML file rather than local Oracle ILOM service processor. Must be followed by = and the pathname to the file.	file.xml

Modify DNS Information

To modify DNS information, use the `ilomconfig modify dns` command. When you specify an XML file name, the command modifies information defined in the exported service processor configuration XML file rather than modifying Oracle ILOM.

- To modify DNS information, type:

```
# ilomconfig modify dns [--nameservers=nameserverlist] [--autodns=enabled|disabled] [--retries=retries] [--searchpath=searchpathlist] [--timeout=timeout] [--xmlfile=filename.xml]
```

Option	Description	Example
--nameservers	List of DNS nameserver IP addresses for Oracle ILOM separated by commas.	10.168.1.10
--auto-dns	Oracle ILOM Auto-DNS state.	enabled or disabled
--searchpath	List of search suffixes in preferred order and separated by commas.	domain1.com, domain2.com
--retries	Number of retry attempts for DNS.	Integer between 0 and 5.
--timeout	Number of seconds to wait for a DNS response. This can be used with up to six search suffixes, each separated by a comma.	2
--xmlfile	Modify specified XML file rather than local Oracle ILOM service processor. Must be followed by = and the pathname to the file.	file.xml

Modify Clock Information

To modify clock information, use the `ilomconfig modify clock` command. When you specify an XML file name, the command modifies information defined in the exported service processor configuration XML file rather than modifying Oracle ILOM.

- To modify clock information, type:

```
# ilomconfig modify clock [--datetime=datetime] [--  
timezone=timezone] [--usntp=enabled|disabled [-ntp-  
server1=ntpserver1] [--ntp-server2=ntpserver2] [--  
xmlfile=filename.xml]
```

Option	Description	Example
<code>--datetime</code>	Oracle ILOM date in <i>MMDDhhmmYYYY</i> format or <i>MMDDhhmmYYYY.ss</i> format.	032514272010
<code>--timezone</code>	Oracle ILOM clock time zone, such as GMT.	enabled or disabled
<code>--usntp</code>	Oracle ILOM NTP client state.	enabled or disabled
<code>--ntp-server1</code>	Oracle ILOM NTP server 1 IP address.	<i>aaa.bbb.ccc.ddd</i>
<code>--ntp-server2</code>	Oracle ILOM NTP server 2 IP address.	<i>aaa.bbb.ccc.ddd</i>
<code>--xmlfile</code>	Modify specified XML file rather than local Oracle ILOM service processor. Must be followed by = and the pathname to the file.	<i>file.xml</i>

Configuring the Host-to-ILOM Interconnect

The Host-to-ILOM Interconnect enables you to communicate locally with Oracle ILOM from the host operating system (OS) without the use of a network management connection (NET MGT) to the server. For more information, see [Host-to-ILOM Interconnect](#).

- [Enable the Host-to-ILOM Interconnect](#)
- [Disable the Host-to-ILOM Interconnect](#)
- [Modify the Host-to-ILOM Interconnect](#)
- [List the Host-to-ILOM Interconnect Settings](#)
- [Verify the Host-to-ILOM Interconnect Settings](#)

Enable the Host-to-ILOM Interconnect

Note:

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

- To enable the Host-to-ILOM Interconnect, type:

```
# ilomconfig 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

Disable the Host-to-ILOM Interconnect

To disable the Host-to-ILOM Interconnect, use the `ilomconfig disable interconnect` command.

- To disable the Host-to-ILOM interconnect, type:

```
# ilomconfig disable interconnect
```

Modify the Host-to-ILOM Interconnect

To modify the Host-to-ILOM Interconnect between the host and Oracle ILOM, use the `ilomconfig modify interconnect` command. This works only when the interconnect is enabled. At least one option must be specified.

- To modify the Host-to-ILOM Interconnect, type:

```
# ilomconfig modify 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

Option	Description	Example
--hostipaddress	Host IP address. This address must be in the format: 169.254.x.x	169.254.175.72

List the Host-to-ILOM Interconnect Settings

To list the interconnect state and IP settings on both the Oracle ILOM and host side of the interconnect, use `ilomconfig list interconnect`.

- To list the Host-to-ILOM interconnect settings, type:

```
# ilomconfig list interconnect
```

Verify the Host-to-ILOM Interconnect Settings

To verify if the Host-to-ILOM Interconnect is up and running do the following:

- To verify the Host-to-ILOM interconnect settings, type:

```
# ilomconfig list interconnect
```

The following is example output for this command.

```
Interconnect
=====
State: enabled
Type: USB Ethernet
SP Interconnect IP Address: 169.254.182.76
Host Interconnect IP Address: 169.254.182.77
Interconnect Netmask: 255.255.255.0
SP Interconnect MAC Address: 02:21:28:57:47:16
Host Interconnect MAC Address: 02:21:28:57:47:17
```

- Make sure that you can ping the SP Interconnect IP Address. For example:

```
# ping 169.254.182.76
```

Managing Oracle ILOM and Host Watchdog

Starting with Oracle Hardware Management Pack 2.3, Oracle ILOM and host watchdog tools are available for systems running a Linux OS.

Note:

Starting with Oracle Hardware Management Pack 2.4.9.0, the HMP watchdog agent and associated ILOM and host watchdog services are no longer available.

- [HMP Watchdog Overview](#)
- [ILOM Watchdog ilomconfig Commands](#)
- [Host Watchdog ilomconfig Commands](#)
- [Troubleshooting ILOM Watchdog and Host Watchdog States](#)

HMP Watchdog Overview

This section provides a brief overview of the HMP Watchdog Agent and the associated ILOM watchdog and host watchdog services.

 **Note:**

Starting with Oracle Hardware Management Pack 2.4.9.0, the HMP watchdog agent and associated ILOM and host watchdog services are no longer available.

- [HMP Watchdog Agent Overview](#)
- [ILOM Watchdog Overview](#)
- [Host Watchdog Overview](#)

HMP Watchdog Agent Overview

The HMP Watchdog Agent is an optional Oracle Hardware Management Pack component that you can install using the Oracle Hardware Management Pack installer. For information on how to install this component, refer to the *Oracle Hardware Management Pack 2.4 Installation Guide*.

Note the following important information regarding the HMP Watchdog Agent:

- Your system must meet the following requirements to run the HMP Watchdog Agent:
 - A Linux operating system installed
 - Oracle Hardware Management Pack 2.3 or later installed
 - Oracle ILOM 3.2.2 or later on the SP
- The agent is not started by default. It must be started after it is installed.
- The agent is not started automatically after host reset (even if it was started before reset). It must be restarted after the host is reset.
- The host or ILOM watchdog configuration is preserved after the host is reset.

The HMP Watchdog Agent provides two services:

- **ILOM watchdog:** If enabled, the ILOM watchdog periodically queries for Oracle ILOM's health. If Oracle ILOM becomes unresponsive, the host performs the specified timer action, which is either Warning or Reset. For more information, see [ILOM Watchdog Overview](#).
- **Host watchdog:** If enabled, the host watchdog watches over the host's health. If the host becomes unresponsive, Oracle ILOM performs the specified timer action, which is one of the following: Warning, Reset, Power Off, or Power Cycle. For more information, see [Host Watchdog Overview](#).

This document covers how to configure the HMP Watchdog Agent services using the `ilomconfig` CLI command.

Information on configuring the HMP Watchdog Agent manually is available at [Configuring the HMP Watchdog Agent](#) in *Oracle Hardware Management Pack 2.4 Server Management Agents User's Guide*.

ILOM Watchdog Overview

The ILOM watchdog enables the host to perform user-settable predefined actions when the host notices that Oracle ILOM has stopped responding.

You can control ILOM watchdog's parameters from the `ilomconfig` command, or by editing the HMP Watchdog Agent's configuration file.

See [ILOM Watchdog Parameters](#) for more information on the ILOM watchdog parameters that can be modified.

When ILOM watchdog determines that Oracle ILOM has stopped responding and therefore performs the timer action, it logs an appropriate message into the HMP watchdog log file, and the host's system log at `/var/log/messages`.



Note:

Using the `ilomconfig` command is the preferred method for controlling ILOM watchdog.

- [ILOM Watchdog Parameters](#)

ILOM Watchdog Parameters

The ILOM watchdog parameters can be set by the user in one of the following ways:

- Preferred method: Use the `ilomconfig modify ilomwatchdog` command with the `ilomwatchdog` target to change the parameters. See [ILOM Watchdog ilomconfig Commands](#).
- Manually edit the configuration file. See [Manually Edit the Configuration File](#) in *Oracle Hardware Management Pack 2.4 Server Management Agents User's Guide*.

The following table lists the ILOM watchdog parameters.

Parameter Name	Parameter in Configuration File	Description	Possible Values	Default value
Query Time Interval	<code>ilom_watchdog_query_time_interval</code>	The periodic time interval, in seconds, at which the host will send a query to Oracle ILOM. If Oracle ILOM does not respond, then this indicates that Oracle ILOM is unresponsive, and the host performs the Timer Action.	Positive integer up to 214748364	60 seconds

Parameter Name	Parameter in Configuration File	Description	Possible Values	Default value
Timer Action	<code>ilom_watchdog_timer_action</code>	The action which the host performs when it notices that Oracle ILOM is unresponsive.	Warning Reset	Warning
Number of Consecutive SP Resets	<code>ilom_watchdog_number_sp_reset</code>	If the value of Timer Action is Reset, this parameter tells the host the number of times the host resets the SP until it gives up.	Positive integer up to 20	2
Admin State	<code>ilom_watchdog_admin_state</code>	The state which the user (Admin) sets the ILOM watchdog.	Enable Disable	Disable

Host Watchdog Overview

Host watchdog gives Oracle ILOM the ability to perform user-settable predefined actions when Oracle ILOM notices that the host has stopped responding.

You can control the host watchdog parameters using the `ilomconfig` command, or by editing the HMP Watchdog Agent's configuration file.



Note:

Using `ilomconfig` command is the preferred method for controlling the host watchdog.

Host watchdog is built on top of standard IPMI watchdog timer functionality. The HMP Watchdog Agent interacts with the IPMI watchdog timer as follows:

- When the host watchdog is enabled by the user, it first checks to see if IPMI watchdog timer is already started. If the IPMI watchdog timer is started, host watchdog issues a log message, indicating that watchdog timer is already started and host watchdog remains in the Disable state.
This addresses the case where someone has already set the IPMI watchdog timer.
- While the host watchdog is enabled, it periodically resets the IPMI watchdog timer to the configuration values of host watchdog.
This addresses the case where someone changes the configuration outside of HMP Watchdog Agent.
- The host watchdog periodically resets the IPMI watchdog timer. When the OS hangs, and therefore the IPMI watchdog timer is not reset before the timer expires, Oracle ILOM performs the action specified by the Timer Action parameter.

See [Host Watchdog Parameters](#) for more information on the host watchdog parameters that can be modified.

- [Host Watchdog Parameters](#)

Host Watchdog Parameters

The following host watchdog parameters can be set by the user in one of the following ways:

- Preferred method: Use the `ilomconfig modify hostwatchdog` command to change the parameters. See [Host Watchdog ilomconfig Commands](#).
- Manually edit the configuration file. See [Manually Edit the Configuration File in Oracle Hardware Management Pack 2.4 Server Management Agents User's Guide](#).

Property Name	Parameter in Configuration File	Description	Possible values	Default value
Timer Action	<code>host_watchdog_timer_action</code>	The action Oracle ILOM takes when the host become unresponsive.	Warning Reset PowerOff PowerCycle	Warning
Timer Value	<code>host_watchdog_timer_value</code>	The periodic time interval, in seconds, at which Oracle ILOM performs the Timer Action if the host is unresponsive.	Positive integer with a maximum of 6553	300
Admin State	<code>host_watchdog_admin_state</code>	The state in which the user (Admin) sets the host watchdog.	Enable Disable	Disable
Reset Period	<code>host_watchdog_reset_period</code>	The periodic time interval, in seconds, at which the host informs Oracle ILOM that the host is alive. This value has to be less than that of Timer Interval.	Positive integer that is less than the value of the Timer Value	60

ILOM Watchdog ilomconfig Commands

The following procedures describe how to use the `ilomconfig` command with an `ilomwatchdog` target to view and configure Oracle ILOM watchdog parameters. Options specific to each command are listed with the command. For general CLI tool options, see [ilomconfig Command Syntax](#).

Note:

Starting with Oracle Hardware Management Pack 2.4.9.0, the HMP watchdog agent and associated ILOM and host watchdog services are no longer available.

- [List Current ILOM Watchdog Settings](#)

- [Enable ILOM Watchdog](#)
- [Disable ILOM Watchdog](#)
- [Modify Current ILOM Watchdog Settings](#)
- [Reset ILOM Watchdog Parameters to Factory Defaults](#)

List Current ILOM Watchdog Settings

- To list the current ILOM watchdog settings, type:

```
# ilomconfig list ilomwatchdog
```

The following is an example of the output of this command:

```
Query time interval: 60 seconds
Timer Action: Warning
Number of consecutive SP resets: 2
Admin state : Disable
Current state: Disable
```

Enable ILOM Watchdog

- To enable ILOM watchdog, type:

```
# ilomconfig enable ilomwatchdog
```

This sets the `ilom_watchdog_admin_state` parameter to Enable.

Disable ILOM Watchdog

- To disable ILOM watchdog, type:

```
# ilomconfig disable ilomwatchdog
```

This sets the `ilom_watchdog_admin_state` parameter to Disable.

Modify Current ILOM Watchdog Settings

- To modify current ILOM watchdog settings, type:

```
# ilomconfig modify ilomwatchdog --option
```

You can change more than one parameter with the same command.

The following options are available for this command.

Option	Possible values	Description
<code>--timer-action= <i>action</i></code>	Warning Reset	This sets the <code>ilom_watchdog_timer_action</code> parameter, which determines the action the ILOM watchdog takes when Oracle ILOM is not responsive.
<code>--number-sp-reset= <i>numspreset</i></code>	Positive integer up to 20	This sets the <code>ilom_watchdog_number_sp_reset</code> parameter, which determines the number of times the SP can be reset consecutively.

Option	Possible values	Description
<code>--query-interval=query interval</code>	Positive integer up to 2147483647	This sets the <code>ilom_watchdog_query_time_interval</code> parameter, which determines the number of seconds between querying for Oracle ILOM's health.

Reset ILOM Watchdog Parameters to Factory Defaults

- To reset ILOM watchdog configuration to factory defaults, type:

```
# ilomconfig reset ilomwatchdog-config
```

See [ILOM Watchdog Parameters](#) for information on defaults for ILOM watchdog parameters.

Host Watchdog ilomconfig Commands

The following procedures describe how to use the `ilomconfig` command with a `hostwatchdog` target to view information about host watchdog and modify the watchdog configuration parameters. Options specific to each command are listed with the command. For general CLI tool options, see [ilomconfig Command Syntax](#).



Note:

Starting with Oracle Hardware Management Pack 2.4.9.0, the HMP watchdog agent and associated ILOM and host watchdog services are no longer available.

- [List Current Host Watchdog Settings](#)
- [Enable Host Watchdog](#)
- [Disable Host Watchdog](#)
- [Modify Current Host Watchdog Settings](#)
- [Reset Host Watchdog Parameters to Factory Defaults](#)

List Current Host Watchdog Settings

- To list `hostwatchdog` settings, type:

```
# ilomconfig list hostwatchdog
```

The following is an example of the output of this command:

```
Timer Action: Warning
Timer Value: 300 seconds
Remaining timer value: 300 seconds
Reset period: 60 seconds
Admin state : Disable
Current state: Disable
```

See [Host Watchdog Parameters](#) for more information on the parameters.

Enable Host Watchdog

- To enable the host watchdog, type:

```
# ilomconfig enable hostwatchdog
```

This sets the `host_watchdog_admin_state` parameter to Enable.

Disable Host Watchdog

- To disable the host watchdog, type:

```
# ilomconfig disable hostwatchdog
```

This sets the `host_watchdog_admin_state` parameter to Disable.

Modify Current Host Watchdog Settings

- To modify current host watchdog settings, type:

```
# ilomconfig modify hostwatchdog --option
```

You can change more than one parameter with the same command.

The following options are available for this command.

Option	Possible Values	Description
<code>--timer-action=action</code>	Warning Reset PowerOff PowerCycle	This sets the <code>host_watchdog_timer_action</code> parameter, which determines the action the host watchdog takes when the timer value expires.
<code>--timer-value=timervalue</code>	Positive integer up to 6553	This sets the <code>host_watchdog_timer_value</code> parameter, which determines the number of seconds before the timer expires.
<code>--reset-period=queryinterval</code>	Positive integer that is less than the value of timer-value parameter	This sets the <code>host_watchdog_reset_period</code> parameter, which determines the number of seconds before resetting the timer value.

Reset Host Watchdog Parameters to Factory Defaults

- To reset host watchdog configuration to factory defaults, type:

```
# ilomconfig reset hostwatchdog-config
```

See [Host Watchdog Parameters](#) for information on defaults for host watch dog parameters.

Troubleshooting ILOM Watchdog and Host Watchdog States

A list of error codes for the `ilomconfig` commands using `ilomwatchdog` and `hostwatchdog` targets are in [ilomconfig Error Codes](#).



Note:

Starting with Oracle Hardware Management Pack 2.4.9.0, the HMP watchdog agent and associated ILOM and host watchdog services are no longer available.

This section describes errors that might occur when the state of the daemon is not what is normally expected by the user.

- [Start and Stop Errors](#)
- [Errors Related to ILOM Watchdog State](#)
- [Errors Related to Host Watchdog State](#)

Start and Stop Errors

The errors shown in the following table appear when the HMP Watchdog Agent has not been started or has stopped. You might see these messages when you run any `ilomconfig` commands with the `hostwatchdog` or `ilomwatchdog` targets.

Error Message	Cause	Solution
ERROR: HMP watchdog agent is not started.	HMP Watchdog Agent has not started.	Run the following command to start HMP watchdog: <pre>service hmp_watchdogd start</pre>
ERROR: HMP watchdog agent is stopped.	HMP Watchdog Agent has been stopped by SIGSTOP. Note: It is dangerous to suspend the operation of the HMP Watchdog Agent by sending it the SIGSTOP signal. It is possible that host watchdog is already enabled, and if the HMP Watchdog Agent is suspended long enough such that host watchdog is not able to reset the IPMI watchdog timer, the IPMI watchdog timer can expire, and Oracle ILOM performs the timer action which can reset the host.	Send a SIGCONT to the HMP Watchdog Agent's process ID as follows: <ol style="list-style-type: none"> 1. Get the HMP Watchdog Agent's process ID from the <code>/var/run/sun-ssm-hmp-watchdogd.pid</code> file. 2. Run the following command: <pre>kill -CONT PID</pre>where <code>PID</code> is the process ID that you identified in Step 1.

Errors Related to ILOM Watchdog State

The errors shown in the following table occur when the ILOM watchdog Admin state (which is set by the user) is different from the Current state . The error message is shown, followed by the reason for the message, and a solution to the problem. In these cases, `ilomconfig list` displays a string indicating the error condition.

Error Message	Problem	Solution
Oracle ILOM does not support health query. Need to have ILOM 3.2.2 or greater.	Oracle ILOM watchdog requires Oracle ILOM 3.2.2 or later to support a health query by IPMI. If you have an earlier version of Oracle ILOM, ILOM watchdog cannot be enabled.	<p>Do one of the following:</p> <ul style="list-style-type: none"> • Disable ILOM watchdog with the following command: <pre>ilomconfig disable ilomwatchdog</pre> • Update Oracle ILOM to a later version. <p>After updating Oracle ILOM, run the following command to reload the watchdog:</p> <pre>/etc/init.d/ hmp_watchdogd reload</pre> <p>or</p> <pre>service hmp_watchdogd reload</pre>
IPMI command failed connection. Check IPMI driver.	ILOM watchdog uses IPMI to communicate with Oracle ILOM. If there is problem with IPMI on the host so that it is unable to communicate with Oracle ILOM, and the ILOM watchdog is currently in Enable state, the ILOM watchdog automatically notices this and changes to Inactive state.	<p>Troubleshoot the IPMI driver on the host.</p> <p>Once the IPMI problem is resolved, the ILOM watchdog detects it during the time of query interval (default is 60 seconds) after the IPMI problem is resolved, and changes to Enable state.</p>
SP is not running Oracle ILOM.	ILOM watchdog is enabled, but the SP is not running Oracle ILOM. If ILOM watchdog is currently in Enable state, the ILOM watchdog automatically notices this and changes to Inactive state.	<p>Reboot the SP.</p> <p>When ILOM watchdog is in inactive state, once the SP is successfully rebooted, and ILOM watchdog successfully sends a health query to Oracle ILOM, it automatically changes to Enable state.</p>
SP is not running Oracle ILOM or Host-to-ILOM Interconnect is not enabled.	On a platform with Emulex Pilot 3 BMC, and ILOM watchdog enabled, either the SP is not running Oracle ILOM or Host-to-ILOM Interconnect is not enabled. If ILOM watchdog is currently in Enable state, the ILOM watchdog automatically notices this and changes to Inactive state.	<p>Make sure Host-to-ILOM Interconnect is enabled, if so, make sure that the SP is running Oracle ILOM.</p> <p>When ILOM watchdog is in inactive state, once the Host-to-ILOM interconnect is enabled, the SP is successfully running Oracle ILOM, and the ILOM watchdog successfully sends a health query to Oracle ILOM, it will automatically change to Enable state.</p>
Unknown failure in ILOM watchdog.	ILOM watchdog is enabled but encounters some unexpected errors	Report the failure to Oracle.

Errors Related to Host Watchdog State

The errors shown in the following table occur when the host watchdog state is affected by the state of the IPMI watchdog timer. The error message is shown, followed by the reason for the message, and a solution to the problem.

Error Message	Host Watchdog State	Problem	Solution
IPMI watchdog timer is enabled outside of Hardware Management Pack.	Admin state : Disable Current state: Enable	The host watchdog has been disabled by the user, but the IPMI watchdog timer is enabled outside of Hardware Management Pack. While in this state (Admin state is Disable and Current State is Enable) if you enable host watchdog, host watchdog will remain in Disable state.	In order to enable hostwatchdog, IPMI watchdog timer needs to be turned off with the following command: <code>ipmitool mc watchdog off</code>
IPMI watchdog timer is enabled outside of Hardware Management Pack.	Admin state : Enable Current state: Enable	First, the Admin state is Enable and the user uses ipmitool to disable the IPMI watchdog timer, then the user uses ipmitool to enable IPMI watchdog timer.	After the duration of the reset period (default is 60 seconds), the host watchdog automatically corrects the situation.
IPMI watchdog timer is disabled outside of Hardware Management Pack.	Admin state : Enable Current state: Disable	The IPMI watchdog timer is disabled outside of Hardware Management Pack while the host watchdog is in Enabled state.	After the reset period (default is 60 seconds), the host watchdog corrects this and re-enable the timer.

Error Message	Host Watchdog State	Problem	Solution
Need to have Oracle ILOM 3.2.2 or greater.	Varies	Oracle ILOM must be version 3.2.2 or greater to run host watchdog.	Do one of the following: <ul style="list-style-type: none"> Disable host watchdog with this command: <pre>ilomconfig disable host watchdog</pre> Upgrade Oracle ILOM to 3.2.3. or greater, then run one of the following commands: <pre>/etc/init.d/ hmp_watchdog reload</pre> or <pre>service hmp_watchdog reload</pre>
IPMI command failed connection. Check IPMI driver.	Varies	Failure sending IPMI request to SP.	Check the IPMI driver on the host side.
IPMI watchdog command failed.	Varies	IPMI watchdog command got to the SP, but encountered failure.	Make sure the IPMI stack is working properly on Oracle ILOM.
Can not determine Oracle ILOM version.	Varies	Host watchdog is unable to determine the Oracle ILOM version.	Issue some IPMI commands to the SP to see if the IPMI commands work on the SP.

8

Using nvmeadm to Configure an NVM Express Device

The `nvmeadm` utility collects and modifies the NVMe device configuration. This utility supports NVMe add-in PCIe cards and NVMe SSDs beginning with the Oracle Flash Accelerator F160 PCIe Card and the 1.6 TB SSD. For a list of supported controllers and servers, see the support matrix at: <http://www.oracle.com/goto/ohmp>.

- [nvmeadm Command Overview](#)
- [List NVMe Controllers](#)
- [List NVMe Namespaces](#)
- [List the Supported LBA Format](#)
- [List NVMe Controller Log Pages](#)
- [List NVMe Features of the Controller](#)
- [Format All Namespaces on the Controller](#)
- [Erase All Namespaces](#)
- [Export an SSD Disk Configuration](#)
- [Import an SSD Disk Configuration](#)

nvmeadm Command Overview

The `nvmeadm` commands use the following syntax:

```
nvmeadm subcommand [option] [controller_name]
```






Note:

If a controller name is not specified for a command, the required information for all controllers is returned.

When a command fails, it returns one of several failure codes listed in [nvmeadm Error Codes](#).

The `nvmeadm` command supports the subcommands listed in the following table.

Subcommand	Function
<code>list</code>	Lists information for the specified controller.
<code>namespace</code>	Lists information for the namespaces of the specified controller.

Subcommand	Function
getlog	Lists NVMe log pages of the controller. There are three log pages: SMART/Health, Error code information, and vendor-specific log information.
getfeature	Lists NVMe features of the controller.
format	<p>Low-level formats specified namespaces, which changes the LBA (Logical Block Address) and metadata size for the controller. All data is destroyed after a low level format.</p> <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> Note:</p> <p>Stop all IO to the NVMe device before attempting to format it. This is not necessary if you are simply obtaining format details using the format <code>-l</code> or <code>--list</code> option.</p> </div>
erase	<p>Erases the NVMe namespace media for the controller.</p> <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> Note:</p> <p>Stop all IO to the NVMe device before attempting this action.</p> </div>
export	Exports the SSD configuration to a file. This file should not be edited or modified.
import	<p>Imports block size and metadata size configuration from a file.</p> <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> Note:</p> <p>Stop all IO to the NVMe device before attempting this action.</p> </div>

The `nvmeadm` command supports the options shown in the following table.

Short Option	Long Option	Subcommands Used With	Description
-?	--help	All	Displays usage information.
-V	n/a	All	Displays version information.
-a	--all	format erase	Selects all namespaces on the controller.

Short Option	Long Option	Subcommands Used With	Description
-b	--blocksize	format	Specifies the LBA data size of a namespace on the controller. This option requires an argument for blocksize. Supported block sizes depend on the controller (see List the Supported LBA Format).
-e	--error	getlog	Retrieves the extended error information.
-f	--format	format	Formats the NVM namespace media on the controller to the logical block size and metadata size specified by -b, -m, -a and -n options.
-f	--filename	export, import	Writes the data to or reads the data from the specified file name.
-h	--health	getlog	Retrieves the SMART/health information of the controller. The information is over the life of the controller and is retained across power cycles.
-l	--list	format	Lists the LBA formats supported by the controller. Each LBA format contains LBA size and metadata size.
-m	--metadatasize	format	Specifies the metadata size of a namespace on the controller. This option requires an argument for metadata size. Supported metadata sizes depend on the supported block sizes (see List the Supported LBA Format).
-n	--namespace	format, namespace, erase	Selects the namespace on the controller.

Short Option	Long Option	Subcommands Used With	Description
-s	--secure	erase	Securely erases all data on the controller. <i>This option was introduced in Oracle Hardware Management Pack 2.4.</i>
-s	--vendor_specific	list, getlog	Retrieves the vendor-specific information of the specified controller. This option is used with the <code>list</code> and <code>getlog</code> subcommands. <ul style="list-style-type: none"> When used with <code>list</code>, this option displays vendor information about the controller. When used with <code>getlog</code>, this option generates <code>nlog</code> and <code>eventlog</code> files for Intel NVMe devices, and generates crash dump and memory dump files for Samsung NVMe devices. This option requires an argument that specifies the directory to which the log pages will be saved.
-v	--verbose	list, namespace	Displays detailed information of a controller or device/namespace, based on the invoked subcommand.

List NVMe Controllers

- To list NVMe controllers in the system, do one of the following:
 - To list all NVMe controllers on the host, type:

```
# nvmeadm list
```

SUNW-NVME-1
 - To list all NVMe controllers with details, type:

```
# nvmeadm list -v

SUNW-NVME-1
PCI Vendor ID:          1111
Serial Number:         1111111111
Model Number:          11111111111111
Firmware Revision:     1.1.1
Number of Namespaces:  1
```

- To list details for a specific controller, type:

```
# nvmeadm list -v controller_name
```

For example:

```
# nvmeadm list -v SUNW-NVME-1

SUNW-NVME-1
PCI Vendor ID:          1111
Serial Number:         1111111111
Model Number:          11111111111111
Firmware Revision:     1.1.1
Number of Namespaces:  1
```

- To list vendor-specific details for a specific controller, type:

```
# nvmeadm list -s controller_name
```

For example:

```
# nvmeadm list -s SUNW-NVME-1

SUNW-NVME-1
  PCI Vendor ID:          0x8086
  PCI Device ID:          0x0953
  PCI Subsystem Vendor ID: 0x108e
  PCI Subsystem ID:       0x370b
  Oracle Part Number:     7090698
  Oracle Model Number:    IFDPC5EA3ORC1.6T
```

List NVMe Namespaces

- To list an NVMe namespace, type:

```
# nvmeadm namespace [-n] [namespace] [-v] [controller_name]
```

For example:

- To list namespaces on all NVMe controllers:

```
# nvmeadm namespace
```

```
SUNW-NVME-1
  Namespace: 1
```

- For details on namespace 1 on controller SUNW-NVME-1:

```
# nvmeadm namespace -n 1 -v SUNW-NVME-1
```

```
SUNW-NVME-1
  Namespace: 1
  Block Size:          512
  Capacity:             786146787328
```

```
Metadata Size:          0
Block Device Name:     /dev/rdisk/c5t0d0s2
```

List the Supported LBA Format

- To list the supported LBA formats on an NVMe controller, type:

```
# nvmeadm format -l [controller_name]
```

For example:

```
# nvmeadm format -l SUNW-NVME-1
```

```
SUNW-NVME-1
  LBA Format: 1
    Block Size:          512
    Metadata Size:      0
  LBA Format: 2
    Block Size:          512
    Metadata Size:      8
  LBA Format: 3
    Block Size:          512
    Metadata Size:      16
  LBA Format: 4
    Block Size:          4096
    Metadata Size:      0
  LBA Format: 5
    Block Size:          4096
    Metadata Size:      8
  LBA Format: 6
    Block Size:          4096
    Metadata Size:      64
```

List NVMe Controller Log Pages

There are three NVMe controller log pages as follows:

- SMART/health information** is gathered over the life of the controller and is retained across power cycles. It includes critical warnings about the controller and device status, such as temperature threshold, available spare, device life status, and various I/O statistics used for calculating I/O performance.
- Error information** is extended error information for commands. A number, which designates the error ID, must be specified with this command.
- Vendor Log information** is the vendor-specific NVMe log implementation. Use this log when working with Oracle Service to troubleshoot errors.
- Do one of the following:
 - To list SMART/health information, type:

```
# nvmeadm getlog -h [controller_name]
```

The following is an example for the `nvmeadm getlog -h` command:

```
# nvmeadm getlog -h SUNW-NVME-1
```

```
SUNW-NVME-1
SMART/Health Information:
```

```

Critical Warning: 0
Temperature: 300 Kelvin
Available Spare: 100 percent
Available Spare Threshold: 10 percent
Percentage Used: 0 percent
Data Unit Read: 0x746da4 of 512k bytes.
Data Unit Written: 0x2d0 of 512k bytes.
Number of Host Read Commands: 0xeacba
Number of Host Write Commands: 0x27
Controller Busy Time in Minutes: 0x0
Number of Power Cycle: 0x10d
Number of Power On Hours: 0x3c8
Number of Unsafe Shutdown: 0xfa
Number of Media Errors: 0x0
Number of Error Info Log Entries: 0x0

```

- To list error information, type:

```
# nvmeadm getlog -e error_id [controller_name]
```

- To save vendor log information to a file, type:

```
# nvmeadm getlog -s directory [controller_name]
```

For example, for a controller named SUNW-NVME-1:

```
# nvmeadm getlog -s /logs SUNW-NVME-1
```

- For a Samsung controller, the files `crashdump_SUNW-NVME-1` and `memorydump_SUNW-NVME-1` are generated and placed under `/logs`.
- For an Intel controller, the files `eventlog_SUNW-NVME-1` and `nlog_SUNW-NVME-1` are generated and placed under `/logs`.

List NVMe Features of the Controller

- To list NVMe features of the controller, type:

```
# nvmeadm getfeature [controller_name]
```

For example:

```
# nvmeadm getfeature SUNW-NVME-1
```

```

SUNW-NVME-1
  Command Arbitration:
    Arbitration Burst: 0
    Low Proirity Weight: 0
    Medium Priority Weight: 0
    High Priority Weight: 0
  Power State: 0
  Temperature Threshold: 358 Kelvin
  Time Limited Error Recovery: 0 of 100 milliseconds
  Number of I/O submission queues allocated: 30
  Number of I/O completion queues allocated: 30
  Interrupt Coalescing Aggregation Time: 0 of 100 micro seconds
  Interrupt Coalescing Configuration:
    Interrupt Vector: 0
    Coalescing Disable: NO
  Write Atomicity Required: YES

```

Format All Namespaces on the Controller

Stop all IO to the NVMe device before attempting format it.

The controller does not support the format of a single namespace. Use the `-a` option to confirm the format of all namespaces. For supported metadata and block size information, see [List the Supported LBA Format](#).

▲ Caution:

All data is destroyed after a low-level format.

- To format all namespaces on a controller, type:

```
# nvmeadm format -f -a -m metadata_size -b block_size
controller_name
```

For example:

```
# nvmeadm format -f -a -m 0 -b 4096 SUNW-NVME-1
```

Erase All Namespaces

Stop all IO to the NVMe device before attempting this action.

The controller does not support the erase of a single namespace. Use the `-a` option to confirm the erase of all namespaces.

▲ Caution:

All data will be destroyed after an erase.

- Choose one of the erase options:

- To erase all namespaces, type:

```
# nvmeadm erase -a controller_name
```

For example:

```
# nvmeadm erase -a SUNW-NVME-1
```

- To securely erase all namespaces, type:

```
# nvmeadm erase -s -a controller_name
```

For example:

```
# nvmeadm erase -s -a SUNW-NVME-1
```

Export an SSD Disk Configuration

- To export an SSD disk configuration to a file, type:

```
# nvmeadm export -f filename.xml controller_name
```

For example:

```
# nvmeadm export -f format.xml SUNW-NVME-1
```

 **Note:**

The exported XML file should not be edited or modified.

Import an SSD Disk Configuration

Stop all IO to the NVMe device before attempting this action.

Only block size and metadata size information can be imported.

- To import an SSD disk configuration from a file, type:

```
# nvmeadm import -f filename.xml controller_name
```

For example:

```
# nvmeadm import -f format.xml SUNW-NVME-2
```

9

Using raidconfig to Configure RAID

`raidconfig` uses a general-purpose cross-OS storage management library to configure RAID volumes using an XML file.

- [raidconfig Command Overview](#)
- [Listing Controller, RAID and Disk Information](#)
- [Creating and Deleting RAID Volumes](#)
- [Adding and Removing Disks and RAID Volumes](#)
- [Modifying a RAID Volume or Controller](#)
- [Starting or Stopping a Task on a Disk or RAID](#)
- [Restoring or Clearing a RAID Controller Configuration](#)
- [Exporting or Importing a RAID Volume Configuration](#)
- [Creating RAID Volumes With Partial Disks](#)

raidconfig Command Overview

This section covers the following information:

- [raidconfig Features](#)
- [raidconfig Requirements](#)
- [raidconfig Command Syntax](#)

raidconfig Features

`raidconfig` allows you to explore, monitor, and configure storage resources connected to the system.



Note:

To use `raidconfig` on storage in a system, the controller that the storage is connected to must support RAID. For a list of supported controllers, see the support matrix at: <https://www.oracle.com/goto/ohmp>.

`raidconfig` provides the following functions:

- Shows, creates, deletes, and modifies RAID volumes.
- Facilitates scripting by using command-line options.
- Configures many similar and dissimilar platforms in a data center.

- Displays the current RAID configuration and writes it to an XML file so it can be edited and used to configure the same or a different platform.
- Represents a logical disk in a portable manner.
For example, using a unique enumeration per controller, instead of a SAS address, facilitates moving the XML file to other platforms.
- Provides a super-set of all configuration options provided by the Adaptec and LSI CLI commands.
- Uses capability checking for particular adapters based on data retrieved from the API.
- Creates nested RAID volumes depending on the controller.

raidconfig Requirements

Before running `raidconfig`, note the following requirements:

▲ Caution:

`raidconfig` can scan your controllers and connected disks and list disks that are either already in a RAID volume, or available to be included in a RAID volume. However, `raidconfig` cannot tell if an available disk has data on it, or if a disk is otherwise used as either a boot disk or logical disk for an application.

Before using `raidconfig` to create volumes (which will overwrite any existing data), use operating system tools to take an inventory of attached disks, their enumeration, and whether they contain data that you want to preserve.

- Root permissions are required to run `raidconfig` commands on Unix-based platforms.
- On Oracle Solaris, `raidconfig` is not compatible with the `raidctl` CLI tool. `raidconfig` supports SAS2 and SAS3, but the `raidctl` tool does not.
- For servers running Oracle Solaris, after hot-plugging any device, run the `devfsadm -C` command to reenumerate all of the system device nodes before running the `raidconfig` command.

raidconfig Command Syntax

The `raidconfig` commands use the following command syntax:

```
raidconfig subcommand target|task -option(s)
```

When a command fails, it returns one of several failure codes listed in [raidconfig Error Codes](#).

The options shown in the following table apply to all CLI Tools commands including `raidconfig`.

Short Option	Long Option	Description
-?, -h	--help	Displays help information.
-V	--version	Displays the tool version.
-q	--quiet	Suppresses informational message output and only returns error codes.
-y	--yes	Confirms operation. Does not prompt user for confirmation on the operation when running.

The `raidconfig` command requires subcommands unless used with the `--help` or `--version` options.

The following table lists the `raidconfig` subcommands.

Subcommand	Function
<code>list</code>	Lists information on controllers, RAID volumes and disks, including disks not in a RAID volume. Specific devices can be selected for display.
<code>create</code>	Creates a RAID volume.
<code>delete</code>	Deletes a RAID volume.
<code>add</code>	Adds a specified disk or spare.
<code>remove</code>	Removes a specified disk or spare.
<code>modify</code>	Modifies a RAID volume or a disk.
<code>start</code>	Starts a maintenance task.
<code>stop</code>	Stops a maintenance task.
<code>restore</code>	Finds the RAID configuration saved on a disk and restores it.
<code>clear</code>	Clears the RAID configuration saved on the disks of a defined controller.
<code>export</code>	Generates an XML file from a RAID configuration.
<code>import</code>	Reads in a RAID configuration from an XML file and creates RAID volumes and spares.

Whenever devices (controllers, RAID volumes, and disks) are used with commands, they must be uniquely identified. For information on how to do so, see the device-naming scheme at [CLI Tools Device-Naming Convention](#).

Device naming is shared with other CLI Tools based on the storage library.

Listing Controller, RAID and Disk Information

This section covers the following information:

- [list Subcommand Overview](#)
- [Display a Brief Listing of All Devices](#)
- [Display a Brief Listing of a Device](#)
- [Display a Detailed Listing of a Device](#)

list Subcommand Overview

The `list` subcommand displays controller, RAID volume, and disk data. The device targets for the `raidconfig list` are listed in the following table.

Target	Description
<code>all</code>	Shows details on all controllers, physical disks, and RAID volumes.
<code>controller</code>	Shows details on all controllers.
<code>disk</code>	Shows the physical disks.
<code>raid</code>	Shows all RAID details.

The `raidconfig list` command supports options listed in the following table.

Short Option	Long Option	Description
<code>-c</code>	<code>--controller</code>	Shows details about a particular controller. This option is followed by the controller ID string.
<code>-r</code>	<code>--raid</code>	Shows details about a particular RAID volume. This option is followed by the RAID ID string.
<code>-d</code>	<code>--disks</code>	Shows details about particular disk(s). This option is followed by a comma-separated list of the disk ID strings.
<code>-v</code>	<code>--verbose</code>	Lists all fields. By default, a brief listing shows only a subset of the fields.

The following data is displayed. Items marked with an asterisk (*) show a brief listing; all other items show a verbose listing.

Controllers:

- Node ID
- Manufacturer*
- Model*
- Part number
- Firmware(F/W) version*
- Serial Number
- RAID Volumes*
- Disks*
- Disks in use by another controller
- PCI address
- PCI vendor ID
- PCI device ID
- PCI subvendor ID

- PCI subdevice ID
- Battery backup status
- Maximum RAID volumes
- Maximum disks per RAID volume
- Supported RAID levels
- Maximum dedicated spares
- Maximum global spares
- Stripe size minimum
- Stripe size maximum
- Disable Auto Rebuild

Disks:

- ID*
- Chassis ID*
- Slot ID*
- Node ID
- Mapped to host OS (true/false)
- Device
- Disabled (true/false)
- In use by another controller
- RAID ID*
- Status*
- Type*
- Media*
- Manufacturer
- Model
- Size
- Serial number
- NAC name
- Spare state (global, dedicated, or N/A)*
- Current task
- Stoppable tasks
- Startable tasks
- Task state
- Task completion percent

RAID volumes:

- Logical ID (0-based)*
- Node ID

- Device name*
- Name (user assigned)*
- Status*
- RAID level*
- Number of disks*
- Capacity*
- Mounted
- Stripe size
- Leg size
- Read cache
- Write cache
- Current task
- Task state
- Task completion percent
- Stoppable tasks
- Startable tasks
- BIOS Boot Target

Display a Brief Listing of All Devices

- To display a brief listing of all available controllers, RAID volumes, disks in use, and available disks, type:

```
# raidconfig list all
```

The following shows sample output from this command.

```
CONTROLLER c0
=====
Manufacturer  Model      F/W Version  RAID Volumes  Disks
-----
Adaptec       0x0285     5.2-0        4              8

RAID Volumes
=====
ID      Name              Device      Status      Num Disks  Level  Size
(GB)
-----
---
c0r0    0919XF5017-0     /dev/sda   OK          1          Simple 146
c0r1    raid1             /dev/sdb   OK          2          0      293
c0r2    raid2             /dev/sdc   OK          3          10     146
c0r3    noname            /dev/sdd   OK          2          0      293

DISKS In Use
=====
ID      Chassis  Slot  RAID ID  Status  Type  Media  Spare  Size
(GB)
-----
```

```

---
c0d0  0      0      c0r0    OK      sas    HDD    -      146
c0d1  0      1      c0r2    OK      sas    HDD    -      146
c0d2  0      2      c0r3    OK      sas    HDD    -      146
c0d3  0      3      c0r3    OK      sas    HDD    -      146
c0d4  0      4      c0r2    OK      sas    HDD    -      146
c0d5  0      5      c0r2    -       sas    HDD    Dedicated 146
c0d6  0      6      c0r1    OK      sas    HDD    -      146
c0d7  0      7      c0r1    OK      sas    HDD    -      146

```

The following table lists the possible RAID statuses that can be displayed with the `raidconfig list all` command.

Status	Meaning
OK	The status of the RAID volume is OK.
DEGRADED	The RAID volume has been degraded.
FAILED	The RAID volume has failed.
MISSING	The controller is reporting that it has a RAID volume is configured but the actual configuration settings aren't available. This status is rare.

The following table lists the possible disk statuses that can be displayed with the `raidconfig list all` command.

Status	Meaning
OK	The status of the disk is OK.
OFFLINE	The disk is offline.
FAILED	The disk has failed.
MISSING	The disk has been removed from a RAID.
INIT	The disk has been initialized.
SPARE	The disk is a spare.

Display a Brief Listing of a Device

- To display a brief listing of a device, type:

```
# raidconfig list subcommand option device
```

For example:

```
# raidconfig list disk -d c0d0
```

```

DISKS Available
=====
ID      Chassis  Slot  RAID ID  Status  Type  Media  Spare  Size (GiB)
-----
c0d0    0        0      -        -       sas   HDD    -       279

```

Display a Detailed Listing of a Device

- To show a detailed listing of a device, type:

```
# raidconfig list device option devicename -v
```

For example for a disk:

```
# raidconfig list disk -d=c0d0 -v

Disk c0d0
=====
ID: c0d0
Chassis: 0
Slot: 0
Node ID: PDS:5000cca0257dbac1
Mapped to Host OS: true
Device: 5000CCA0257DBAC0
Disabled: false
Type: sas
Media: HDD
Manufacturer: HITACHI
Model: H106030SDSUN300G
Size (GiB): 279
Serial Number: 001214N74K2B          PQJ74K2B
NAC Name: /SYS/SASBP/HDD0
Current Task: none
```

For example for a controller:

```
# raidconfig list controller -v

CONTROLLER c0
=====
Node ID: mptir2:50:00.0
Manufacturer: LSI Logic
Model: SG-SAS6-INT-Z
F/W Version: 11.05.03.00
Serial Number: 500605b005468020
RAID Volumes: 1
Disks: 8
PCI Address: 50:00.0
PCI Vendor ID: 0x1000
PCI Device ID: 0x0072
PCI Subvendor ID: 0x1000
PCI Subdevice ID: 0x3050
Battery Backup Status: Not installed
Max RAID Volumes: 2
Max Disks per RAID Volume: 256
Supported RAID Levels: 0, 1, 10
Max Dedicated Spares: 0
Max Global Spares: 2
Stripe Size Min (KB): 64
Stripe Size Max (KB): 64
```

The following table lists the possible Battery Backup statuses that can be displayed with the `raidconfig list controller` command.

Status	Meaning
Not Installed	The battery backup option is not installed.
OK	The status of the battery backup is OK.
Charging	The battery backup is charging.
Discharging	The battery backup is discharging.

Status	Meaning
Low voltage	There is low voltage to the HBA on-board memory and the battery backup has become its primary source of power.
High temperature	The battery backup is overheating. This can cause the battery to stop charging and reduce its life expectancy.
Failed	The battery backup has failed and might need to be replaced.
Missing	The battery backup hardware is missing, malfunctioning, unplugged or fully discharged.

Creating and Deleting RAID Volumes

This section covers the following information:

- [Create a RAID Volume](#)
- [Delete a RAID Volume](#)

Create a RAID Volume

Before using `raidconfig` to create volumes (which will overwrite any existing data on selected disks), use operating system tools to take an inventory of attached disks, their enumeration, and whether they contain data that you want to preserve. Be careful not to overwrite your OS boot disk or other logical disks used by applications.

- To create a RAID volume, type:



```
# raidconfig create raid options -d disks
```

For example, to create a RAID 0 volume with a stripe size of 128 Kb and read-ahead caching enabled on controller 1, type the following command:

```
# raidconfig create raid --stripe-size=128 --read-cache=enabled -d c1d0,c1d1
```

The `create raid` subcommand must take the `-d` option in addition to one or more of the options shown in the following table.

Short Option	Long Option	Description
-d	--disks	Specifies a list of disks with a comma separating the disk ID numbers.
N/A	--level	Specifies the RAID level of the volume e.g. 0, 1, 1E, 5, 10, 50, 60 etc. The levels supported for a particular controller can be seen in the controller 'Supported RAID Levels' field from the list command. If this option is not supplied, a level of '0' is used.
N/A	--name	Assigns the user-defined name that identifies the RAID volume. This name can be set to an empty string ("").

Short Option	Long Option	Description
N/A	--read-cache	<p>Read cache can be one of the following:</p> <ul style="list-style-type: none"> disabled – Disables RAID read caching enabled – Enables RAID read ahead caching enabled_adaptive – Enables RAID read adaptive caching <div style="border: 1px solid #0070c0; padding: 10px; margin-top: 10px;"> <p> Note:</p> <p>Only supported with SAS2 HBAs starting with the Sun Storage 6 Gb SAS RAID PCIe HBA, Internal (SGX-SAS6-R-INT-Z, SG-SAS6-R-INT-Z) and SAS3 HBAs starting with the Oracle Storage 12 Gb SAS RAID PCIe HBA, Internal (7110116, 7110117).</p> </div>
N/A	--stripe-size	Specifies the stripe size, in kilobytes, of the RAID volume to be created. If this option is not supplied, the controller uses a default size.
N/A	--subarrays	For nested RAID levels (10, 50), specifies the size of the RAID components in number of physical disks.
N/A	--subdisk-size	See Creating RAID Volumes With Partial Disks .
N/A	--write-cache	<p>Write cache can be one of the following:</p> <ul style="list-style-type: none"> disabled – Disables RAID write caching. enabled – Enables RAID write caching. enabled_protect – Enables caching only if the battery is available. <div style="border: 1px solid #0070c0; padding: 10px; margin-top: 10px;"> <p> Note:</p> <p>Only supported with SAS2 HBAs starting with the Sun Storage 6 Gb SAS RAID PCIe HBA, Internal (SGX-SAS6-R-INT-Z, SG-SAS6-R-INT-Z) and SAS3 HBAs starting with the Oracle Storage 12 Gb SAS RAID PCIe HBA, Internal (7110116, 7110117).</p> </div>

The maximum capacity of the RAID volume is not configurable. You can create RAID from partial disks if the HBA or controller support it, and all the disks are of the same size.

Delete a RAID Volume

- To delete a RAID volume, type:
`# raidconfig delete raid option`

For example:

- To delete RAID volume 1 created on controller 1, type:
`# raidconfig delete raid -r c1r1`
- To delete all RAID volumes, type:
`# raidconfig delete raid --all`

The `delete raid` requires one of the options shown in the following table.

Short Option	Long Option	Description
-r	--raid	Deletes the volume listed by ID number.
N/A	--all	Deletes all RAID volumes on all controllers. <code>raidconfig</code> queries the storage management library to determine if the RAID disks have been mounted. If so, it generates a warning message to the user and queries the user to delete the RAID volume.

Adding and Removing Disks and RAID Volumes

This section covers the following information:

- [Add a Disk to a RAID Configuration](#)
- [Remove a Disk from a RAID Volume](#)
- [Add Spare Disks](#)
- [Remove a Spare Disk or a RAID Volume](#)

Add a Disk to a RAID Configuration

The `add disk` subcommand adds a specified disk to a RAID configuration.

Only certain RAID levels, such as RAID 5 or 6, allow disks to be added to their configuration when in a non-degraded (healthy) state. Only RAID levels that support redundancy allow disks to be added.

- To add a specific disk to a RAID volume, type:
`# raidconfig add disk -d disk -r raidvolume`
- For example:
- ```
raidconfig add disk -d c0d2 -r c0r1
```

 **Note:**

If you list the disk's properties after adding a disk, the RAID ID is not updated to reflect that it has been added to a RAID volume until the add process is complete.

The `add disk` subcommand requires the options shown in the following table.

| Short Option | Long Option | Description                                                            |
|--------------|-------------|------------------------------------------------------------------------|
| -d           | --disks     | Specifies the list of disks that you want to add to the RAID volume.   |
| -r           | --raid      | Specifies the RAID volume ID number to which you want to add the disk. |

## Remove a Disk from a RAID Volume

The `remove disk` subcommand removes a disk from a RAID volume. Only RAID levels that support redundancy allow for disks to be removed.

- To remove a specific disk from a RAID volume, type:  

```
raidconfig remove disk -d disk -r raidvolume
```

For example:

```
raidconfig remove disk -d c0d0 -r c0r1
```

This subcommand requires the options shown in the following table.

| Short Option | Long Option | Description                                                          |
|--------------|-------------|----------------------------------------------------------------------|
| -d           | --disks     | Specifies the disk that you want to remove from the RAID volume.     |
| -r           | --raid      | Specifies the RAID volume ID from which you want to remove the disk. |

## Add Spare Disks

The `add spare` subcommand adds global or dedicated spare disks.

- To create two global spares using the specified disks, type:

```
raidconfig add spare -d disk,disk
```

For example:

```
raidconfig add spare -d c1d0,c1d1
```

- To create two dedicated spares on a RAID volume using the specified disks, type:

```
raidconfig add spare -d disk,disk -r raidvolume
```

For example:

```
raidconfig add spare -d c1d0,c1d1 -r c1r0
```

The `add spare` subcommand requires one of the options shown in the following table.

| Short Option | Long Option | Description                                                                                                                                                                                                                                 |
|--------------|-------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -d           | --disks     | This <i>mandatory option</i> specifies a list of disk ID numbers, separated by commas. If the <code>-r</code> option is not used, the disks are added as global spares.                                                                     |
| -r           | --raid      | Only used when working with dedicated spares. If a RAID Volume ID is specified, the spares should be added as dedicated spares for this RAID Volume. Note that some controllers do not support dedicated spares and the command might fail. |

## Remove a Spare Disk or a RAID Volume

The `remove spare` subcommand removes disks global spares or as dedicated spares on a RAID volume.

- Do one of the following:
  - To remove two disks as global spares, type:
 

```
raidconfig remove spare -d disk,disk
```

For example:

```
raidconfig remove spare -d c1d0,c1d1
```
  - To remove two disks as dedicated spares on a RAID volume, type:
 

```
raidconfig remove spare -d disk,disk -r raidvolume
```

For example:

```
raidconfig remove spare -d c1d0,c1d1 -r c1r0
```

This subcommand requires the options shown in the following table.

| Short Option | Long Option | Description                                                                                                                                               |
|--------------|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------|
| -d           | --disks     | Specifies disks to remove. Disk ID numbers are separated by commas. If the <code>-r</code> option is not defined, the disks are removed as global spares. |
| -r           | --raid      | If a RAID volume ID is specified, the disks should be removed as dedicated spares from this RAID volume.                                                  |

## Modifying a RAID Volume or Controller

This section covers the following information:

- [Modify a RAID Volume](#)
- [Modify a Controller](#)
- [Modify the BIOS Boot Target](#)

- [Disable Auto Rebuild](#)
- [Modify a RAID Volume Name](#)
- [Enable or Disable JBOD Mode](#)

## Modify a RAID Volume

The `modify raid` subcommand modifies the attributes of a RAID volume.

- To modify a RAID volume, type:

```
raidconfig modify raid -r raidvolume option
```

For example:

```
raidconfig modify raid -r c0r0 --write-cache=disabled
```

The `modify raid` subcommand requires the option shown in the following table.

| Short Option | Long Option | Description                                                                                        |
|--------------|-------------|----------------------------------------------------------------------------------------------------|
| -r           | --raid      | Specifies the RAID volume to modify. This is required for the <code>modify raid</code> subcommand. |

The following table lists additional options for the `modify raid` subcommand.

| Option                  | Description                                                                                                                                                                                         |
|-------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| --name                  | Specifies the user-defined name to identify the RAID volume. Can be set to an empty string ("").                                                                                                    |
| --read-cache            | Write cache can be one of the following:<br>disabled – Disables RAID read caching<br>enabled – Enables RAID read caching<br>enabled_adaptive – Enables RAID read adaptive caching                   |
| --write-cache           | Write cache can be one of the following:<br>disabled – Disables RAID write caching.<br>enabled – Enables RAID write caching.<br>enabled_protect – Enables caching only if the battery is available. |
| --bios-boot-target=true | Sets the boot target. When this option is set to true for a specific RAID volume, that RAID volume becomes the BIOS boot target.                                                                    |

## Modify a Controller

The `modify controller` command modifies certain controller attributes.

- To modify a controller, type:

```
raidconfig modify controller -c controller option
```

For example:

```
raidconfig modify controller -c c1 --disable-auto-rebuild=true
```

The `modify controller` subcommand requires the option shown in the following table.

| Short Option | Long Option  | Description                                                                                             |
|--------------|--------------|---------------------------------------------------------------------------------------------------------|
| -c           | --controller | Specifies the controller to modify. This is required for the <code>modify controller</code> subcommand. |

The following table lists an additional option for the `modify controller` subcommand.

| Option                            | Description                                                                                                                                                                                                                                                   |
|-----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| --disable-auto-rebuild=true false | Disables auto rebuild. When this option is set to true for a specific controller, auto rebuild will be disabled. If the option is set to false, a hot spare can automatically replace a faulty disk, in which case a long running background task is started. |

 **Note:**

Not all controllers support modifications of `--disable-auto-rebuild`.

## Modify the BIOS Boot Target

The RAID volume with ID 0 is the default boot target. If you want to change the boot target, use the `--bios-boot-target` option.

- To change the bios boot target, type:

```
raidconfig modify raid -r raidvolume --bios-boot-target=true
```

For example:

```
raidconfig modify raid -r c0r0 --bios-boot-target=true
```

## Disable Auto Rebuild

When a hotspare disk replaces a faulty disk, it will start autobuilding the volume to use the hotspare disk if auto rebuild is enabled. If you do not want to start the long-running background task automatically, you can disable this feature.

- To disable auto rebuild, type:

```
raidconfig modify controller -c controller id --disable-auto-rebuild=true
```

For example:

```
raidconfig modify controller -c c0 --disable-auto-rebuild=true
```

## Modify a RAID Volume Name

To modify the user-specified name of a RAID volume:

- To change the user-specified name of a RAID volume, type:

```
raidconfig modify raid -r raidvolume --name name
```

For example:

```
raidconfig modify raid -r c0r0 --name engineering
```

## Enable or Disable JBOD Mode

Some SAS3 Host Bus Adapters (HBA) with hardware RAID support include an option to enable JBOD mode. JBOD mode allows the operating system to access a disk directly without first creating a RAID volume on it. If JBOD mode is not enabled, the operating system is not able to see the disk until the disk is included in a RAID volume.

If supported by your HBA, you can enable JBOD mode on either a disk or controller. If you enable JBOD mode on the controller, all the disks on that controller will be in JBOD mode. Disk JBOD mode cannot be enabled until its controller has JBOD mode enabled. Check your HBA documentation to see if it supports JBOD mode.



### Note:

Do not disable JBOD mode on a controller if one of its disks in JBOD mode has the OS installed on it. Also, do not disable JBOD mode on an individual disk if that disk has the OS installed on it.

- To enable or disable JBOD mode on a controller, type:

```
raidconfig modify controller -c controller --jbod enabled|disabled
```

When you enable JBOD mode on the controller, all the disks on that controller will be in JBOD mode. Disks can then have JBOD mode enabled or disabled individually. If you disable JBOD mode on a controller, any disks in JBOD mode will have JBOD mode disabled.

- To enable or disable JBOD mode on a disk, type:

```
raidconfig modify disk -d disk --jbod enabled|disabled
```

The following example shows output for JBOD mode enabled on c0, then disabled on just disk 7.

```
CONTROLLER c0
=====
Manufacturer Model F/W Version RAID Volumes Disks

LSI Logic MegaRAID 9361-8i 4.220.20-3050 1 8

RAID Volumes
=====
ID Name Device Status Num Disks Level Size
```

```
(GiB)

c0r1 OEL /dev/sda OK 1 0 465

DISKS In Use
=====
ID Chassis Slot RAID ID Status Type Media Spare Size (GiB)

c0d0 0 0 c0r1 OK sata HDD - 465

DISKS Available
=====
ID Chassis Slot RAID ID Status Type Media Spare Size (GiB)

c0d1 0 1 - JBOD sas HDD - 137
c0d2 0 2 - JBOD sas HDD - 137
c0d3 0 3 - JBOD sas HDD - 137
c0d4 0 4 - JBOD sas HDD - 137
c0d5 0 5 - JBOD sas HDD - 137
c0d6 0 6 - JBOD sata HDD - 466
c0d7 0 7 - OK sata HDD - 466
```

## Starting or Stopping a Task on a Disk or RAID

The `start task` and `stop task` subcommands control the execution of maintenance tasks on a disk or RAID volume.

- [Executing Tasks on a Disk or RAID Volume](#)
- [Start or Stop a Task on a Disk or RAID Volume](#)

## Executing Tasks on a Disk or RAID Volume

There are a variety of tasks that can be run on a RAID volume and its disks using the `start task` or `stop task` subcommands.

The available background tasks are shown in the following table.

| Task                 | Description                                                                                                                                                                                                                                                                 |
|----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>verify</code>  | Checks the validity of the RAID volume redundant data.                                                                                                                                                                                                                      |
| <code>init</code>    | Initializes the RAID volume to write out the initial parity values. The initialization goes over the entire volume and initializes the parity data.                                                                                                                         |
| <code>copy</code>    | Copies and moves an online physical disk onto a hotspare or unconfigured good drive. The copy is performed while the volume is online. Once completed, the destination disk is added to the logical volume configuration while the original source disk is removed from it. |
| <code>rebuild</code> | Regenerates the data of a single physical disk that is part of a logical volume with data redundancy. The physical disk is reconstructed from another physical disk and/or parity disks. A disk rebuild typically occurs after a disk replacement or repair.                |
| <code>clear</code>   | Clears a physical disk by writing zeroes over the entire disk.                                                                                                                                                                                                              |



 **Note:**

Not all devices support all tasks. To check the tasks a device supports, use the `list` subcommand and check the output under Startable tasks. If this field is blank, the device does not support any tasks.

The `start task` and `stop task` subcommands accept the options shown in the following table.

| Short Option | Long Option | Description                                                                                                                                                              |
|--------------|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| -t           | --task      | Specifies the type of task to execute. Possible options are <code>verify</code> , <code>init</code> , <code>rebuild</code> , <code>clear</code> , or <code>copy</code> . |
| -d           | --disk      | Specifies the disk to execute the task on. Required by the <code>rebuild</code> and <code>clear</code> tasks.                                                            |
| -r           | --raid      | Specifies the RAID volume to execute the task on. Required by the <code>verify</code> and <code>init</code> tasks.                                                       |
| n/a          | --src-disk  | Specifies the source disk to use in a <code>copy</code> task.                                                                                                            |
| n/a          | --dst-disk  | Specifies the destination disk to use in <code>copy</code> task.                                                                                                         |

## Start or Stop a Task on a Disk or RAID Volume

The `start task` and `stop task` subcommands control the execution of maintenance tasks on a disk or RAID volume. For more information on available tasks and options, see [Executing Tasks on a Disk or RAID Volume](#).

- To start or stop a task on a disk or RAID volume, type:
 

```
raidconfig start task -t taskname [-d|-r]
```

 or
 

```
raidconfig stop task -t taskname [-d|-r]
```

The following are command examples for the `start task` and `stop task` subcommands:

- A RAID ID must be provided for the `verify` check (`verify`) and initialization task (`init`).
  - To start the `verify` task on a specified RAID volume, type:
 

```
raidconfig start task -t verify -r=raidvolume
```

 For example:
 

```
raidconfig start task -t verify -r=c0r1
```
  - To stop the `init` task on a specified RAID volume, type:
 

```
raidconfig stop task -t init -r=raidvolume
```

 For example:

```
raidconfig stop task -t init -r=c0r1
```

- A disk must be provided for the rebuild and clear tasks.

- To start the `rebuild` task on a specified disk, type:

```
raidconfig start task -t rebuild -d=disk
```

For example:

```
raidconfig start task -t rebuild -d=c0d1
```

 **Note:**

This can only be run on a disk that is part of a RAID volume.

- To start the `clear` task on a specified disk, type:

```
raidconfig start task -t clear -d=disk
```

For example:

```
raidconfig start task -t clear -d=c0d1
```

 **Note:**

This can only be run on a disk that is not part of a RAID volume.

- Source and destination disks must be provided for the copy task.

To start the `copy` task from one disk to another, type:

```
raidconfig start -task -t copy --src-disk=source_disk --dst-disk=destination_disk
```

For example:

```
raidconfig start -task -t copy --src-disk=c0d2 --dst-disk=c0d3
```

 **Note:**

The source disk must be in a RAID volume. The destination disk cannot be in a RAID volume.

## Restoring or Clearing a RAID Controller Configuration

This section covers the following information:

- [Check to See If a Controller Configuration Exists](#)
- [Restore a RAID Controller Configuration](#)
- [Clear a RAID Controller Configuration](#)

## Check to See If a Controller Configuration Exists

1. To determine if an old configuration exists on the disks, view the controller's verbose properties. Type:
 

```
raidconfig list controller -v
```

 The controller's properties are listed.
2. View the property Disks In Use by Another Controller.
  - a. If the Disks In Use by Another Controller property is set to True, then an old configuration exists. This can be either restored or cleared.
  - b. If the Disks In Use by Another Controller property is set to False, then an old configuration does not exist.

### Note:

If an old configuration does not exist and you attempt to run the `restore config` or `clear config` subcommands, `raidconfig` displays an error.

## Restore a RAID Controller Configuration

The `restore config` subcommand finds a RAID configuration stored on disks and restores this configuration to the destination controller.

- To restore a RAID configuration saved on disks to a defined controller, type:

```
raidconfig restore config -c=controller_id
```

where *controller\_id* is the controller the RAID configuration is restored to.

The `restore config` subcommand requires the options shown in the following table.

| Short Option | Long Option  | Description                  |
|--------------|--------------|------------------------------|
| -c           | --controller | Specifies the controller ID. |

## Clear a RAID Controller Configuration

The `clear config` subcommand finds a RAID configuration stored on disks and removes the configuration.

- To clear a RAID configuration saved on disks, type:

```
raidconfig clear config -c=controller_id
```

where *controller\_id* is the controller the RAID configuration is cleared from.

The `clear config` subcommand requires the options shown in the following table.

| Short Option | Long Option  | Description                  |
|--------------|--------------|------------------------------|
| -c           | --controller | Specifies the controller ID. |

## Exporting or Importing a RAID Volume Configuration

This section covers the following information:

- [Export a RAID Volume Configuration](#)
- [Import a RAID Volume Configuration](#)

### Export a RAID Volume Configuration

The `export` subcommand writes XML-formatted configuration or inventory data to a file. Inventory data is a snapshot of all the fields for the controllers, RAID volumes, and disks. Configuration data contains only attributes that can be set and imported onto another system to configure that system's RAID volumes in the same manner.

The `export` subcommand requires a file name as a modifier. If a file by that name exists, the tool prompts to overwrite the file (unless the `-y` option is used). If the hyphen (-) is given for the filename, then the XML-formatted configuration is written to the screen.

- To export the inventory or a configuration and write it to a file, do one of the following:
  - To export the inventory data and write it to a file, type:
 

```
raidconfig export inventory filename.xml
```
  - To export a configuration and write it to a file, type:
 

```
raidconfig export config filename.xml
```

This subcommand requires at least one of the types shown in the following table.

| Option    | Description                                                                                         |
|-----------|-----------------------------------------------------------------------------------------------------|
| inventory | Exports and writes all controller, RAID volume, and physical disk information to an XML file.       |
| config    | Exports and writes only configuration fields that can be imported to another system to an XML file. |

### Import a RAID Volume Configuration

The `import` subcommand reads an XML-formatted configuration file and configures RAID volumes based on the file. If the creation of a specific RAID volume fails, the error is logged and the next RAID volume in the file is created.

The `import` subcommand requires the `config` type and a file name for the XML file.

#### Note:

You cannot import a configuration into a system if the configuration includes disks that are already defined in a RAID volume or as a spares.

- To configure the RAID volumes according to a configuration file, type:

```
raidconfig import config filename.xml
```

## Creating RAID Volumes With Partial Disks

The `--subdisk-size` option is available for the `raidconfig create` command to define the size of RAID volumes. This option is used to define the size of the partial disks to be used in a RAID volume.

- [Guidelines for Using the RAID Volume Size Option](#)
- [Disk Display](#)
- [Partial Disk Properties in XML File](#)
- [Create a RAID Volume with Partial Disks](#)
- [Adding or Removing a Partial Disk](#)

## Guidelines for Using the RAID Volume Size Option

Keep the following guidelines in mind when using the RAID volume `--subdisk-size`:

- The total sizes for the RAID volumes designated in the `--subdisk-size` option cannot exceed the available size for any of the disks. The total size can be less than or equal to the disk size, but it cannot be larger.
- You cannot create a RAID volume using a partial disk on a disk that is configured as part of a RAID volume. Once a disk has been included in a RAID volume, the disk is marked as "In Use" and cannot be used to create another RAID volume, even if just a part of the disk is used.

For example, the following sequence of commands is not allowed:

```
raidconfig create raid --disk=c0d0,c0d2 --subdisk-size=50
raidconfig create raid --disk=c0d0,c0d2 --subdisk-size=100
```

The second command results in an error.

- When creating multiple RAID volumes at the same time using the `--subdisk-size` option, all of the RAID volumes are configured with the same name if the `--name` option is used.

If this occurs, you can rename the volumes using the `raidconfig modify` command.

- You can delete a RAID volume on a partial disk, but if the partial disk is used in another RAID volume, the disk will be marked as "In Use". You will not be able to create another RAID volume using that disk.

## Disk Display

The `list all` subcommand indicates that a disk is part of more than one RAID volume. A row is added for each disk/raid combination under the `DISKS In Use` list.

The `Size` column shows the size of the subdisk used to create the RAID volume.

The following is an example of the DISKS In Use output:

```
DISKS In Use
=====
ID Chassis Slot RAID ID Status Type Media Spare Size (GiB)

c0d0 0 17 c0r0 OK sas HDD - 50
c0d0 0 17 c0r1 OK sas HDD - 100
c0d0 0 17 c0r2 OK sas HDD - 200
c0d2 0 18 c0r0 OK sas HDD - 50
c0d2 0 18 c0r1 OK sas HDD - 100
c0d2 0 18 c0r2 OK sas HDD - 200
```

## Partial Disk Properties in XML File

If a RAID volume was created using partial disks, `raidconfig` stores the size of the sub-disk in the XML output generated by the `export` command. An example of a disk property is shown below:

```
<disk>
<chassis_id>0</chassis_id>
<slot_id>1</slot_id>
<subdisk_size>100</subdisk_size>
</disk>
```

## Create a RAID Volume with Partial Disks

Use the `--subdisk-size` option with the `raidconfig create` to create a RAID volume with partial disks:

- To create a RAID volume with partial disks, type:

```
raidconfig create raid --disk=disks --subdisk-size=sizes
```

For example, the following command creates three RAID volumes with subdisks within disks `c0d0` and `c0d2` sized at 50, 75, and 100 GB:

```
raidconfig create raid --disk=c0d0,c0d1 --subdisk-size=50,75,100
```

```
Create RAID level 0 volumes using disk sizes 50, 75, 100 from the
following disk(s):
Disk c0d0 (controller 0 slot 0)
Disk c0d1 (controller 0 slot 1) [y/n]? y
RAID created successfully
```

If the `--subdisk-size` option is not used, the `raidconfig create` command creates a single RAID volume from the defined disks.

## Adding or Removing a Partial Disk

The `add` and `remove` subcommands are supported for partial disks. If a disk contains multiple RAID volumes, they can be added and removed. For information on using the `raidconfig add` and `remove` commands, see [Adding and Removing Disks and RAID Volumes](#).

 **Note:**

When the disk supports multiple RAID volumes, only use the first RAID volume in the `add` and `remove` commands.

An example of removing a disk is shown below:

```
raidconfig remove disk -r=c0r4 -d=c0d0
```

```
Removing the following disk(s) from RAID c0r4:
```

```
Disk c0d0 (controller 0 slot 0) [y/n]? y
```

```
Successfully removed disk from RAID
```

```
raidconfig list all
```

```
CONTROLLER c0
```

```
=====
```

Manufacturer	Model	F/W Version	RAID Volumes	Disks
LSI Logic	0x0079	2.130.353-1803	6	7

```
RAID Volumes
```

```
=====
```

ID	Name	Device	Status	Num Disks	Level	Size (GiB)
c0r0	0	c3t0d0p0	OK	1	0	558
c0r1		c3t1d0p0	OK	1	0	278
c0r2		c3t2d0p0	OK	1	0	136
c0r3		c3t3d0p0	OK	1	0	70
c0r4		c3t4d0p0	DEGRADED	2	1	50
c0r5		c3t5d0p0	DEGRADED	2	1	100

```
DISKS In Use
```

```
=====
```

ID	Chassis	Slot	RAID ID	Status	Type	Media	Spare	Size (GiB)
c0d1	0	1	c0r4	OK	sas	HDD	-	50
c0d1	0	1	c0r5	OK	sas	HDD	-	100
c0d3	0	3	c0r0	OK	sas	HDD	-	558
c0d4	0	4	c0r1	OK	sas	HDD	-	278
c0d5	0	6	c0r3	OK	sas	HDD	-	70
c0d6	0	7	c0r2	OK	sas	HDD	-	136

```
DISKS Available
```

```
=====
```

ID	Chassis	Slot	RAID ID	Status	Type	Media	Spare	Size (GiB)
c0d0	0	0	-	OK	sas	HDD	-	279
c0d2	0	2	-	OK	sas	HDD	-	279

The following is an example of adding a disk:

```
raidconfig add disk -r=c0r4 -d=c0d2
```

```
Adding the following disk(s) to RAID c0r4:
Disk c0d2 (controller 0 slot 2) [y/n]? y
Successfully added disk to RAID
```

```
raidconfig list all
```

```
CONTROLLER c0
=====
```

Manufacturer	Model	F/W Version	RAID Volumes	Disks
LSI Logic	0x0079	2.130.353-1803	6	7

```
RAID Volumes
=====
```

ID	Name	Device	Status	Num Disks	Level	Size (GiB)
c0r0	0	c3t0d0p0	OK	1	0	558
c0r1		c3t1d0p0	OK	1	0	278
c0r2		c3t2d0p0	OK	1	0	136
c0r3		c3t3d0p0	OK	1	0	70
c0r4		c3t4d0p0	DEGRADED	2	1	50
c0r5		c3t5d0p0	DEGRADED	2	1	100

```
DISKS In Use
=====
```

ID	Chassis	Slot	RAID ID	Status	Type	Media	Spare	Size (GiB)
c0d1	0	1	c0r4	OK	sas	HDD	-	50
c0d1	0	1	c0r5	OK	sas	HDD	-	100
c0d2	0	2	c0r4	INIT	sas	HDD	-	50
c0d2	0	2	c0r5	INIT	sas	HDD	-	100
c0d3	0	3	c0r0	OK	sas	HDD	-	558
c0d4	0	4	c0r1	OK	sas	HDD	-	278
c0d5	0	6	c0r3	OK	sas	HDD	-	70
c0d6	0	7	c0r2	OK	sas	HDD	-	136

```
DISKS Available
=====
```

ID	Chassis	Slot	RAID ID	Status	Type	Media	Spare	Size (GiB)
c0d0	0	0	-	OK	sas	HDD	-	279



# 10

## Using ubiosconfig to Update the UEFI BIOS

`ubiosconfig` provides a CLI tool for configuring BIOS on Oracle x86 servers that support UEFI BIOS. For legacy x86 systems, use the `biosconfig` tool. See [Using biosconfig to Update the BIOS](#).

`ubiosconfig` enables you to save server UEFI BIOS settings to an XML file, then load the settings from the XML file to configure UEFI BIOS settings on another server. For more information on UEFI BIOS, see your server documentation.

For information on the systems supported for each tool, refer to the support matrix at:

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

- [ubiosconfig Command Overview](#)
- [Export UEFI Settings to an XML File](#)
- [Import UEFI BIOS Settings to a Server](#)
- [Display Information on Changes to UEFI BIOS Settings](#)
- [Cancel Pending Changes to UEFI BIOS Settings](#)
- [Reset the UEFI BIOS Settings to Factory Default](#)

### ubiosconfig Command Overview

This section covers the following information:

- [ubiosconfig Features](#)
- [ubiosconfig and Service Processor Access](#)
- [ubiosconfig Command Syntax](#)

### ubiosconfig Features

`ubiosconfig` command line utility allows you to update the UEFI BIOS configuration on supported Oracle x86 servers. The `ubiosconfig` command can be used to get current configuration settings or set configuration settings. When used to get configuration settings, `ubiosconfig` generates XML output showing the configuration. When used to set configuration settings, `ubiosconfig` reads XML input describing the configuration settings.

The `ubiosconfig` commands can be directed at a local or remote Oracle ILOM service processor, or an XML configuration file. This file can then be used as a golden image to make changes to multiple Oracle ILOM service processors. You can either export the configuration of the UEFI BIOS or create a new XML configuration file.

The XML tag definitions are determined by the current system BIOS. These values can vary by system type and it is not recommended that you use the same XML file to update the BIOS configuration across different system types.

 **Caution:**

Setting the BIOS configuration incorrectly may result in undesired operation or system instability. Refer to your server documentation before using `ubiosconfig`.

## ubiosconfig and Service Processor Access

When accessing UEFI BIOS configurations on the service processor (SP), `ubiosconfig` can be used over a local Host-to-ILOM interconnect or a remote Ethernet network connection as follows:

- When using local access, `ubiosconfig` uses the fastest local interface available. If a Host-to-ILOM connection is available this fast connection is used, otherwise the slower KCS interface is used. See [Host-to-ILOM Interconnect](#).

 **Note:**

For systems with an Oracle ILOM version earlier than 3.2.4, you must manually include credentials using the `-H` and `-U` options (described below) for any commands that access a service processor. If credentials are not provided the commands will default to the slower local KCS interface to access the local service processor.

- When using remote Ethernet network access, `ubiosconfig` must present login credentials using a command line argument (SP hostname and user account with root access as described in [Command Options for Accessing Oracle ILOM Over a Remote Network Connection](#)). In addition, command execution over a remote network connection is encrypted using the TLS protocol. This means that a client-side trusted SSL certificate for the Oracle ILOM SP being accessed must be present on the host to validate the connection. This certificate checking feature is the default for a remote network connection when using the `fwupdate`, `ilomconfig` and `ubiosconfig` commands.
- [Obtaining SSL Certificates for TLS Access](#)
- [Command Options for Accessing Oracle ILOM Over a Remote Network Connection](#)

## Obtaining SSL Certificates for TLS Access

In order to use TLS encryption when accessing a Oracle ILOM SP over a remote network connection, a client-side trusted certificate must be available on the host for the Oracle ILOM SP you will be accessing. Note the following:

- Ensure that you've installed the latest TLS and OpenSSL patches for your operating system (Oracle requires TLS 1.2 support at a minimum).
- Oracle Hardware Management Pack commands that perform SSL certificate validation for a remote network connection to a service processor look for hashed symbolic links to client-side trusted certificates in certain directories, depending on the operating system.

- For Linux, the expected directory for certificate hashed links is `/etc/ssl/certs`.
- For Oracle Solaris, the expected directory for certificate hashed links is `/etc/openssl/certs`.

If you opt to install certificates in some other directory, you will need to include a command line argument (as described in [Command Options for Accessing Oracle ILOM Over a Remote Network Connection](#)) that specifies the certificate directory when issuing Oracle Hardware Management Pack commands that perform client-side SSL certificate validation.

To obtain a client-side trusted certificate from a service processor and prepare it for validation, do the following:

1. Obtain a PEM format certificate from the target Oracle ILOM SP. You can use one of the following methods:
  - Use a web browser to log in to the Oracle ILOM SP. If this is the first time logging in to the SP, the browser will prompt you for a security exception at which point you can view and export the certificate in PEM format (`.pem`) to a directory.

If you've already accepted the certificate from a previous browser login, you can export it from the browser's stored servers certificates and export it in PEM format (`.pem`) to a directory.

- You can also run an OpenSSL command from the host to obtain the certificate. For example:

```
echo | openssl s_client -connect sp_ip:623 | sed -n "/--BEGIN/,/--END/ p" > path_to_cert/certname.pem
```

Where `sp_ip` is the host name or IP address of the SP, `path_to_cert` is the directory path to where the certificate will be copied, and `certname` is the file name for the PEM format certificate.

#### Note:

To avoid the possibility of a man-in-the-middle attack, execute this command using a trusted channel or verified using an independent second channel.

- Or, set up your own certification authority and sign a certificate to upload to Oracle ILOM. If you choose to create your own custom certificates, refer to the Oracle ILOM documentation for details.
2. Change ownership of the certificate file you downloaded to `root:root` and file permissions to `-rw-r--r--` (numeric value 644).
  3. Create a hash symbolic link of your downloaded certificate.
    - For Red Hat and Oracle Linux 6 and 7, there is a script called `cacertdir_rehash` which is part of the `authconfig` package. You can use this script to create your hash symbolic link in the certificate directory. For example:

```
/usr/sbin/cacertdir_rehash /etc/ssl/certs
```

Refer to the Red Hat and Oracle Linux documentation for more details.

- For SLES Linux 12, there is a script called `c_rehash` which is part of the OpenSSL package. You can use this script to create your hash symbolic link in the certificate directory. For example:

```
/usr/bin/c_rehash /etc/ssl/certs
```

Refer to the SLES Linux documentation for more details.

- For Oracle Solaris 11, rehash the certificate directory (this will also create the hash symbolic link in the `/etc/openssl/certs` directory). For example:

```
/usr/sbin/svcadm refresh svc:/system/ca-
certificates:default
```

Refer to the Oracle Solaris 11 documentation for more details.

- For Oracle Solaris 10, if you have installed the required patches the `/etc/openssl` directory should be present. Follow these steps to prepare the directory, hash the certificate and create the symbolic link:

- a. Create the `certs` directory under `/etc/openssl` if it does not already exist. Type:

```
mkdir -p /etc/openssl/certs
```

- b. Set the group ownership for the directory. Type:

```
chgrp sys /etc/openssl/certs
```

- c. Set permissions for the directory. Type:

```
chmod 0755 /etc/openssl/certs
```

- d. Create the hash name for your certificate. Type:

```
hashname=`openssl x509 -hash -noout -in path_to_cert/
certname.pem`
```

Where `path_to_cert` is the directory path to your downloaded certificate and `certname` is the file name for the PEM format certificate.

- e. Create the symbolic link for your certificate. Type:

```
ln -s "path_to_cert/certname.pem" "/etc/openssl/
certs/$hashname.0"
```

 **Note:**


The extension `.0` is an example, if there is another certificate using the `.0` extension, use the next available number for your extension.


Refer to the Oracle Solaris 10 and OpenSSL documentation for more details.

4. Ensure that the service processor Common Name (for example, ORACLESP-1000NML000) has been added to the domain name system (DNS) for your network. This name should match the Common Name found in the certificate file.

## Command Options for Accessing Oracle ILOM Over a Remote Network Connection

The credential and certificate options listed in the following table are supported for `ubiosconfig` when accessing a service processor over a network connection. An example of usage follows the table.

Short Option	Long Options	Description
-H	-- remote_hostname= <i>sp_ip</i>	<p>The host name, Common Name, or IP address of the remote service processor as specified by <i>sp_ip</i>. This option must be used in combination with the -U option.</p> <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> <b>Note:</b></p> <p>When accessing an SP over a remote Ethernet network connection, client-side SSL certificate validation is performed by default. For proper validation, you must use the Common Name stored in the client-side SSL certificate and the DNS server for the SP remote host name (e.g. -H ORACLESP-1000NML000). Otherwise, you will receive a "hostname validation failed" error.</p> </div>
-U	-- remote_username= <i>username</i>	<p>The user name with root access used to log in to the remote service processor as specified by <i>username</i>. This option must be used in combination with the -H option.</p>
-t	--intfname= <i>interface</i>	<p>Specifies the IPMI interface to use. No auto-detect is attempted. Supported interfaces that are compiled in are visible in the usage help output (socket interfaces in case -H option is used). See the -T description for more information.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.9.0.</i></p>

Short Option	Long Options	Description
-T	--remote-intfname-fallback= <i>interface</i>	<p>Selects the least secured IPMI socket interface to use if more secure interfaces are not supported. The tool attempts the most secure interface first (orcltIs). If the BMC does not support the interface, then attempt the next most secured socket interface until the specified interface. Supported socket interfaces that are compiled in are visible in the usage help output in the appropriate order. If lanplus or lan is specified, certificate checking is disabled when attempting the orcltIs interface.</p> <div style="border: 1px solid #0070C0; padding: 10px; margin: 10px 0;"> <p> <b>Note:</b></p> <p>If the -T or -t option is not specified, then no auto-detect is enabled and only the orcltIs interface is attempted including certificate checking.</p> </div> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.9.0.</i></p>
n/a	--cert-dir= <i>pathname</i>	<p>Location of trusted certificates as specified by <i>pathname</i>. Use this option if your client-side SSL certificate is in a different directory than the expected default system certificate directory.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.4.</i></p>
n/a	--no-cert-check	<p>Do not perform SSL certificate checking.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.4.</i></p>

For example, where encryption is required for data transmitted over the network, use these command options to execute a command on a service processor over the network:

```
ubiosconfig export all --remote-hostname=sp_ip --remote-username=username --cert-dir=pathname -xml_file=filename.xml
```

where *sp\_ip* in this case is the Common Name for the target system's SP, *username* is the user name with login access rights to perform the operation, *pathname* is the path to the directory that contains your trusted certificate if it is not installed in the expected system certificate directory (see [Obtaining SSL Certificates for TLS Access](#)), and *filename* is the name of the XML file to which you are exporting configurations.

Once your certificate is validated and you are then prompted for the Oracle ILOM user password.

**Note:**

The Oracle ILOM user password required by the network connection can be piped in on stdin for scripting use.

## ubiosconfig Command Syntax

The `ubiosconfig` commands use the following command syntax:

```
ubiosconfig subcommand type [option]
```

If you use the `--help` or `--version` options, the `ubiosconfig` command does not require subcommands; otherwise one or more subcommands are mandatory.


When a command fails, it returns one of several failure codes listed in [ubiosconfig Error Codes](#).

`ubiosconfig` supports the subcommands listed in the following table.


Subcommand	Function
<code>import</code>	Import a configuration XML file that will be applied to the server's UEFI BIOS at next boot.
<code>export</code>	Export the server's UEFI BIOS configuration to a local XML file.
<code>cancel</code>	Cancel pending UEFI BIOS configuration changes.
<code>list</code>	List status information regarding pending UEFI BIOS import or export operations.
<code>reset</code>	Reset the server's UEFI BIOS configuration to factory default at next boot.

The following table lists general options available for `ubiosconfig`. Options specific to each subcommand are described in the section that describes using the subcommand.

Short Option	Long Option	Description
<code>-, -h</code>	<code>--help</code>	Displays help information.

Short Option	Long Option	Description
-H	-- remote_hostname= <i>sp_ip</i>	<p>The host name, Common Name, or IP address of the remote service processor as specified by <i>sp_ip</i>. This option must be used in combination with the -U option.</p> <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> <b>Note:</b></p> <p>When accessing an SP over a remote Ethernet network connection, client-side SSL certificate validation is performed by default. For proper validation, you must use the Common Name stored in the client-side SSL certificate and the DNS server for the SP remote host name (e.g. -H ORACLESP-1000NML000). Otherwise, you will receive a "hostname validation failed" error.</p> </div>
-U	-- remote_username= <i>username</i>	<p>The user name with root access used to log in to the remote service processor as specified by <i>username</i>. This option must be used in combination with the -H option.</p>
-t	--intfname= <i>interface</i>	<p>Specifies the IPMI interface to use. No auto-detect is attempted. Supported interfaces that are compiled in are visible in the usage help output (socket interfaces in case -H option is used). See the -T description for more information.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.9.0.</i></p>



Short Option	Long Option	Description
-T	--remote-intfname-fallback= <i>interface</i>	<p>Selects the least secured IPMI socket interface to use if more secure interfaces are not supported. The tool attempts the most secure interface first (orcltIs). If the BMC does not support the interface, then attempt the next most secured socket interface until the specified interface. Supported socket interfaces that are compiled in are visible in the usage help output in the appropriate order. If lanplus or lan is specified, certificate checking is disabled when attempting the orcltIs interface.</p> <div style="border: 1px solid #0070C0; padding: 10px; margin-top: 10px;"> <p> <b>Note:</b></p> <p>If the -T or -t option is not specified, then no auto-detect is enabled and only the orcltIs interface is attempted including certificate checking.</p> </div> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.9.0.</i></p>
-V	--version	Displays the tool version.
n/a	--cert-dir= <i>pathname</i>	<p>Location of trusted certificates as specified by <i>pathname</i>. Use this option if your client-side SSL certificate is in a different directory than the expected default system certificate directory.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.4.</i></p>
n/a	--no-cert-check	<p>Do not perform SSL certificate checking.</p> <p><i>This option was introduced in Oracle Hardware Management Pack 2.4.4.</i></p>

## Export UEFI Settings to an XML File

The `export` subcommand exports a server's UEFI BIOS settings to an XML file.

- To export UEFI BIOS settings to an XML file, type:

```
ubiosconfig export type -x filename.xml option
```


where *type* is the supported type described below, *filename* is the optional path, and *option* is one of the options described below.

The supported type for export is listed in the following table.

Type	Description
all	Export all current server UEFI BIOS settings.

The supported options for export are listed in the following table. When executing this command over a remote network connection, see [ubiosconfig and Service Processor Access](#).

Short Option	Long Option	Description
-f	--force	Ignore safeguards, and export the BIOS settings to an XML file regardless of current system state.
-x	--xml_file	The path to the target XML file for the current UEFI settings. Without this option, settings are displayed on the screen.
-y	--yes	Bypass any user confirmation prompt when overwriting an existing output file of the same name.

 **Note:**

There is no guarantee of accuracy in the data when using the `--force` option.

## Import UEFI BIOS Settings to a Server

The `import` subcommand imports UEFI BIOS settings stored in an XML file to the server at next boot.

- To import UEFI BIOS settings stored in an XML file, type:


```
ubiosconfig import type -x filename.xml option
```

where *type* is one of the options described below, *filename* is the path to the XML file you want to import settings from and *option* is one of the options described below.

The supported types of import are listed in the following table.

Type	Description
all	Import all options from the XML file to the server's BIOS at next boot.
boot	Import only boot options from the XML file to the server's BIOS at next boot.
config	Import only configuration options from the XML file to the server's BIOS at next boot.

The possible option for the import is listed in the following table. When executing this command over a remote network connection, see [ubiosconfig and Service Processor Access](#).

Short Option	Long Option	Description
-f	--force	Ignore safeguards, and import the BIOS XML file regardless of current system state.   <b>Note:</b> There is no guarantee of accuracy in the data when using the --force option.
-x	--xml_file	The path to the XML file that contains the UEFI settings to import. This is a mandatory option.
-y	--yes	Bypass any user confirmation prompt when overwriting an existing output file of the same name.

## Display Information on Changes to UEFI BIOS Settings

The `list` subcommand in conjunction with type `status` displays information about pending changes to UEFI BIOS settings at the next server boot.

- To display information on UEFI BIOS setting changes, type:

```
ubiosconfig list status
```

## Cancel Pending Changes to UEFI BIOS Settings

The `cancel` subcommand in conjunction with type `config` cancels any pending changes to UEFI BIOS settings.

- To cancel pending changes to UEFI BIOS settings, type:

```
ubiosconfig cancel config
```

## Reset the UEFI BIOS Settings to Factory Default

The `reset` subcommand resets the UEFI BIOS settings to factory defaults at the next server boot.

- To reset the UEFI BIOS settings to factory default, type:

```
ubiosconfig reset type
```

where `type` is one of the supported types listed in the following table. When executing this command over a remote network connection, see [ubiosconfig and Service Processor Access](#).

Type	Description
<code>config</code>	Reset the server's UEFI BIOS to factory defaults at next power cycle. Any pending UEFI BIOS changes from <code>ubiosconfig</code> are added to the factory defaults.

<b>Type</b>	<b>Description</b>
cancel	Cancel any pending reset change to the server's UEFI BIOS settings.

# 11

## Using zoningcli to Configure Zoning for the SPARC T3-1 Server

`zoningcli` provides a CLI tool for configuring zoning on Oracle SPARC T3-1 servers with the following qualifications:

- Runs Oracle Solaris OS.
- Has a 16-disk backplane (SAS-2 expander).
- If your Oracle SPARC T3-1 server is using the two on-board hard disk controllers, the 16-disk backplane *must* be zoned.
- If the server is using the SGX-SAS6-R-INT-Z SAS2 internal PCIe RAID HBA, the 16 disk backplane must *not* be zoned.

`zoningcli` can split zoning into two separate zones:

- Zone A (group 10) has the first 8 disks (PHY 0 - 7) and the first controller (PHY 20 -23)
- Zone B (group 11) has the last 8 disks (PHY 8 - 15) and the second controller (PHY 16-19)

Before attempting to change the server's zoning state, read the product notes pertaining to the 16 disk backplane.

You must have root permission to run `zoningcli`.

- [zoningcli Command Overview](#)
- [List Zoning Information](#)
- [Enable and Disable Zoning](#)
- [Configure Zoning](#)

### zoningcli Command Overview

The `zoningcli` commands use the following command syntax:

```
zoningcli subcommand
```

You must have root permission to run the `zoningcli` command.

The options listed in the following table apply to all CLI Tools commands, including `zoningcli`.

Short Option	Long Option	Description
-?	--help	Help—Displays help information.
-V	--version	Version—Displays the tool version.

If you use the `--help` or `--version` options, the `zoningcli` command does not require subcommands, otherwise one or more subcommands are mandatory.

`zoningcli` supports the subcommands listed in the following table.

Command	Function
<code>enable</code>	Enable the zoning.
<code>disable</code>	Disable the zoning.
<code>list</code>	Show the current zoning information.
<code>config</code>	Set up the expander in a pre-defined zoning configuration.

## List Zoning Information

The following procedure describes how to use the `list` subcommand to check the current zoning information. This enables you to see if zoning is currently enabled or disabled.

- To list zoning information, type:  

```
zoningcli list expander
```

The current zoning information is listed.

## Enable and Disable Zoning

The following procedure describes how to use the `enable zoning` and `disable zoning` subcommands to control zoning. Zoning is set up by default.

- To enable zoning, type:  

```
zoningcli enable zoning
```
- To disable zoning, type the following command:  

```
zoningcli disable zoning
```

## Configure Zoning

Use the `config` subcommand to set up the expander in a pre-defined zoning configuration.

- To set up the expander in a pre-defined zoning configuration, type:  

```
zoningcli config zoning
```

# 12

## CLI Tools Error Codes

The following information is covered in this section.

- [Common Error Codes](#)
- [biosconfig Error Codes](#)
- [fwupdate Error Codes](#)
- [hwmgmtcli Error Codes](#)
- [ilomconfig Error Codes](#)
- [nvmeadm Error Codes](#)
- [raidconfig Error Codes](#)
- [ubiosconfig Error Codes](#)
- [zoningcli Error Codes](#)

### Common Error Codes

The following table lists the common command error codes. Each error code has a string associated with it. The error code is printed to the log file and to the `stdout` file.

Code Number	Error Description
0	OK.
1	Invalid option.
2	Invalid subcommand.
3	Subcommand not supported.
4	Invalid device format.
5	Cannot create XML file.
6	Cannot read XML file.
7	Cannot retrieve application data.
8	Internal error.
9	Insufficient memory.
10	Invalid boolean argument.
11	Option not supported.
12	Storage library initialization failure.
13	Entered name is too long.
14	Invalid name after subcommand.
15	XML filename required.
16	Invalid argument.
17	Failure writing XML file.


Code Number	Error Description
18	Device is busy, command cannot be completed.
19	User terminated by pressing ctrl-c.
20	Insufficient privilege to execute command.
21	One or more arguments are missing.
22	Unsupported XML file. Please see errors.
23	XML parse failure.
24	Cannot find XML file.
25	XML file contains no records.
26	The current directory is not writeable.
27	Invalid type.
28	The prerequisite criteria fails priority requirement.
29	Prerequisite criteria causes forever loop.
30	IPMI timeout. Wait a few sections and try again.
31	Installation issues detected.
32	Platform not supported.
33	Oracle ILOM version not supported.
34	Command cannot be run in virtual environment.
35	Unlink file failure.
36	Mandatory option is required.
37	Operand is required.
38	Subcommand is not presented in command line.
39	Remote options are not supported.
45	Invalid log directory (symbolic link, wrong permission or ownership) or cannot open log file.
50	Cannot connect to BMC interface.
51	Missing <code>-username</code> option.
92	Interface already exists.

## biosconfig Error Codes

The following table lists the `biosconfig` errors and the actions to take when they occur.



Code Number	Description
64	Execute biosconfig as root.

 **Note:**

Do not run more than one instance of `biosconfig` at the same time. There are no locks in place (for any OS) to allow for multiple simultaneous accesses.

## fwupdate Error Codes

The following table lists the `fwupdate` command error codes.

You can also list the error codes using the `fwupdate list error-codes` command. See [list Subcommand Overview](#) for more information.

Code Number	Error Description
200	Invalid device type.
201	Invalid device target type.
202	Invalid device ID, please run "fwupdate list all" to verify id.
203	Reset of component failed.
204	Firmware check failed for component.
205	Firmware download failed for component.
206	Specified component and specified image type do not match.
207	Must specify an image file name when doing an update.
208	Could not read specified image file.
209	Reset of this component type is not supported.
210	Specified component type does not match devices type.
211	Must specify device to update.
212	Update canceled by user.
213	Firmware version information not available. Reset necessary to activate new firmware.
214	Version verification failed.
215	Final version is being reported the same as the starting version. Update may have succeeded, please check update documentation.
216	Missing or corrupt firmware file referenced by firmware metadata file.
217	Metadata file invalid or corrupt.
218	Metadata error. Prerequisite and priority settings conflict.
219	Power control option is not supported for pre-application.
220	Power control option is not supported for post-application.

Code Number	Error Description
221	Power Control option is not supported.
222	Requested component not available.
223	Can't verify version information, no XML provided.
224	Metadata does not include support for this host.
225	Could not identify host type.
226	A valid subcommand required.
227	Invalid option entered.
228	Must specify device to reset.
229	Cannot open file to write XML output.
230	Metadata XML file is required.
231	Invalid priority level entered.
232	Cannot read firmware metadata XML file.
233	Missing required command argument.
234	SP has not recovered in the allotted time limit after an update. The default time for the SP to recover after an update is 15 minutes or a value specified in the metadata XML file.
235	Update forced failure due to a configuration error in the metadata XML file.
236	Oracle ILOM backup image feature is not supported on the target service processor.
254	Thread management failure.

## hwmgmtcli Error Codes

The following table lists the `hwmgmtcli` command error codes.

Code Number	Error Description
242	Initialize HDL library failure.
243	HDL library command failure.
244	Subsystem not supported.

## ilomconfig Error Codes

The following table lists the `ilomconfig` and watchdog timer error codes.



### Note:

More about the ILOM and host watchdog timer is described in the *Server Management Agents User's Guide*.

Code Number	Error Description
50	Cannot connect to BMC interface.
51	Missing <code>-username</code> option.
52	Missing <code>-password</code> option.
53	User already exists.
54	Missing <code>-communityname</code> option.
55	Specified community already exists.
56	User does not exist.
57	Community name does not exist.
58	Delete failed.
59	Failures occurred during restore.
60	Must specify option to modify.
61	No such property.
62	Oracle ILOM login failure.
63	Invalid role value.
64	Invalid permission value.
66	Invalid IP discovery value.
67	Invalid IP state value.
68	Invalid IP address.
69	Invalid auto DNS value.
70	Invalid Use NTP value.
71	Product serial number does not match current system.
72	Oracle ILOM error occurred.
73	Cannot modify interconnect when disabled (use enable command).
74	ILOM not reachable over internal LAN.
75	Credential Failure.
76	Cannot manage interconnect when hostmanaged is set to false.
77	Could not connect to remote SP by LAN with supplied credentials.
78	Specified Command can not be used with a remote connection.
79	Oracle ILOM version does not support LAN over USB.
80	ILOM Interconnect required for fault forwarding.
81	SNMP timeout occurred while setting up fault forwarding.
82	Failed to configure ILOM SNMP correctly.
83	Service Processor has conflicting configuration. Refer to release notes for resolution.
93	Cannot set property.
170	Reading the watchdog configuration file.
171	Writing to the watchdog configuration file.
172	Reading the watchdog state file.

Code Number	Error Description
173	Watchdog query time interval exceeded.
174	Reading watchdog PID file.
175	Watchdog sending SIGHUP.
176	Watchdog IPMI get WDT.
177	Watchdog timer not started.
178	Watchdog timer stopped.
179	Watchdog incompatible reset period timer value.
180	Watchdog timer value maximum.
181	Watchdog ILOM SP reset value maximum.
182	Unable to configure network.

## nvmeadm Error Codes

The following table lists the `nvmeadm` error codes.

Code Number	Error Description
190	Invalid namespace.
191	Invalid controller.
192	Invalid block size and/or metadata size.
193	Command failed on one or more device(s).
194	Invalid directory name.
195	Operation canceled.

## raidconfig Error Codes

Errors might be returned if you attempt to configure the RAID entry for an unsupported parameter. For example, if the RAID controller does not support the configured RAID level, the CLI displays a user-friendly error string identifying the misconfiguration and returns a matching error code.

The following table lists the error codes and strings specific to this tool.

Code Number	Error Description
100	No controllers available.
101	Controller does not support RAID.
102	No physical disks associated with controller.
103	Invalid controller.
104	Invalid disk.
105	Invalid RAID volume.
106	RAID level not supported by controller.
107	Default RAID level not supported.

Code Number	Error Description
108	A defined disk is in use.
109	Number of disks exceeds allowed number for this level.
110	Failure retrieving internal data.
111	Number of disks requested exceeds the number of available disks.
112	Cannot define both actual and requested number of disks.
113	Option not supported by controller.
114	Invalid stripe size for controller.
115	Invalid number of subarrays.
116	Cannot retrieve RAID data.
118	RAID creation failure.
119	RAID deletion failure.
120	Disk defined multiple times.
121	Disks must be in the same controller.
122	The maximum number of RAID Volumes has been created.
123	Invalid RAID configuration.
124	The RAID Volume is in use.
125	Incomplete RAID configuration.
126	Failure writing internal data.
127	Command requires disks to be entered.
128	Disk is not a dedicated spare.
129	Disk is not a global spare.
130	Controller does not support dedicated spares.
131	Controller does not support global spares.
132	Command requires disks or RAID volume to be entered.
133	A defined disk is not in a RAID volume.
134	Cannot set both read and write cache in same command.
135	Import could not create RAID volumes or spares - disks may be in use.
136	Subarrays option is required for this RAID level.
137	Incomplete command, no options have been supplied.
138	Number of disks requested exceeds the number of available disks with the same capacity.
139	RAID configuration does not have enough disks for the requested RAID level.
140	RAID configuration has too many disks for the requested RAID level.
141	Disk detected as in use by another controller. Use raidconfig restore or clear command.
142	The number of spares exceeds the maximum allowed by controller.

Code Number	Error Description
143	This command does not support the number-disks option.
144	Task type is invalid.
145	Task type must be defined.
146	Task type is only valid for disks.
147	Task type is only valid for RAID Volumes.
148	For this task, disk must not be in use.
149	For this task, disk must be in a RAID Volume.
150	Command currently cannot be executed.
151	The source disk must be in a RAID Volume.
152	The destination disk must not be in a RAID Volume.
153	The source and destination cannot be the same disk.
154	The source and destination are not the same size.
155	No foreign configuration detected for controller.
156	Unable to add disk to RAID Volume.
157	Task cannot be started, make sure task is listed in Startable Tasks.
158	Task cannot be stopped, make sure task is listed in Stoppable Tasks.
159	Invalid command, filename must come before options.
160	All disks must be the same size.
161	Command is not valid for this RAID level.
162	Subdisk sizes must be less than disk capacity.
163	Could not restore controller configuration.
164	The maximum number of subdisks is 16.
165	Invalid configuration, make sure spare is same size as disks in RAID Volume.
166	Disk mode is set to JBOD. Disable JBOD mode then try again.
167	Disk is in use by RAID controller. Remove from RAID configuration then try again.

## ubiosconfig Error Codes

The following table lists the `ubiosconfig` errors.

Code Number	Error Description
50	Cannot connect to IPMI bmc device.
84, 85	Cannot update BIOS, update in progress.
86	Invalid configuration file provided.
87	Invalid boot configuration provided.
88	Invalid boot and configuration provided.

---

Code Number	Error Description
89	Failed to update BIOS.
90	BIOS partially updated.
91	BIOS out of sync.
247	UEFI is not supported for system.

## zoningcli Error Codes

The following table lists the `zoningcli` command error codes.

Code Number	Error Description
250	Storage management library failure.
251	Zoning CLI SMP command failure.
252	Zoning CLI running on a not supported platform.
253	The expander does not have zoning support.

# Index

## A

---

### automatic mode

- fwupdate
- update subcommand, [5-25](#)

## B

---

### biosconfig, [4-1](#)

#### boot order

- next boot, [4-7](#)
- overview, [4-6](#)
- persistent, [4-8](#)

#### CMOS configuration, [4-10](#), [4-13](#)

- dynamic setting, [4-14](#)
- individual settings, [4-12](#)
- static setting, [4-13](#)

#### CMOS golden image

- applying, [4-11](#)
- capturing, [4-10](#)

#### device terminology, [4-2](#)

#### error codes, [12-2](#)

#### extraneous output, [4-14](#)

#### options, [4-3](#)

#### Oracle Solaris OS, [4-4](#)

#### overview, [4-1](#)

#### requirements, [4-1](#)

#### view commands, [4-5](#)

#### view version, [4-5](#)

#### XML files, [4-2](#)

### boot order

#### biosconfig

- device, [4-9](#)
- function, [4-9](#)
- next boot, [4-7](#)
- PCI bus, [4-9](#)
- persistent, [4-8](#)

#### methods for changing, [4-7](#)

### boot target

- modify using `raidconfig`, [9-15](#)

## C

---

### certificate checking

- SP access over the network when using `biosconfig`, [10-2](#)
- SP access over the network when using `fwupdate`, [5-5](#)
- SP access over the network when using `ilomconfig`, [7-3](#)

### clock information

- listing, [7-17](#)
- modifying, [7-22](#)

### CMOS

- applying golden image, [4-11](#)
- capturing golden image, [4-10](#)
- configuring dynamic setting, [4-14](#)
- configuring individual settings, [4-12](#)
- configuring static setting, [4-13](#)

### command syntax

- CLI tools common, [3-1](#)

### Common Name

- TLS encryption when using `fwupdate`, [5-5](#)
- TLS encryption when using `ilomconfig`, [7-3](#)
- TLS encryption when using `ubiosconfig`, [10-2](#)

## D

---

### device naming

- CLI tools common, [3-2](#)

### disk

- adding, [9-11](#)
- removing, [9-12](#)

### DNS information

- listing, [7-17](#)
- modifying, [7-21](#)

## E

---

### error codes

- `biosconfig`, [12-2](#)
- common, [12-1](#)
- `fwupdate`, [12-3](#)
- `hwmgmtcli`, [12-4](#)
- `ilomconfig`, [12-4](#)



**error codes (continued)**

- [nvmeadm, 12-6](#)
- [raidconfig, 12-6](#)
- [ubiosconfig, 12-8](#)
- [zoningcli, 12-9](#)

[exporting inventory data, 9-21](#)

**F****fwupdate, 5-1****automatic mode**

- [Oracle ILOM update, 5-30](#)
- [service processor update, 5-30](#)
- [SPARC fallback image, 5-34](#)
- [system firmware update, 5-30](#)

[command overview, 5-10](#)

[description of automatic and manual modes, 5-24](#)

[error codes, 12-3](#)

[execution summary, 5-46](#)

[Linux MMIO access settings and Intel network controllers, 5-3](#)

[list subcommand, 5-13](#)

- [listing all components, 5-18](#)
- [listing specific components, 5-21](#)
- [overview, 5-14](#)

[Oracle ILOM update, 5-36](#)

[overview, 5-1](#)

[Secure Boot settings and Intel storage devices and controllers, 5-3](#)

[service processor update, 5-36](#)

[SP access over the network, 5-5](#)

[update subcommand](#)

- [automatic mode, 5-25](#)
- [single device mode, 5-36](#)

**H****Host-to-ILOM Interconnect**

- [disabling, 7-23](#)
- [enabling, 2-1, 7-23](#)
- [listing settings, 7-24](#)
- [modifying, 7-23](#)

**hwmgmtcli, 6-1**

- [command overview, 6-1](#)
- [error codes, 12-4](#)
- [exporting subsystem information, 6-3](#)
- [listing subsystem information, 6-2](#)
- [viewing open problems, 6-3](#)

**I****ilomconfig, 7-1**

[command usage, 7-8](#)

**ilomconfig (continued)**

- [creating a user, 7-18](#)
- [creating SNMP community, 7-19](#)
- [deleting a user, 7-18](#)
- [error codes, 12-4](#)
- [exporting XML configuration, 7-11](#)
- [features, 7-1](#)
- [importing XML configuration, 7-13](#)
- [IPv4 network settings](#)
  - [listing, 7-16](#)
  - [modifying, 7-19](#)
- [IPv6 network settings](#)
  - [listing, 7-16](#)
  - [modifying, 7-20](#)
- [listing clock information, 7-17](#)
- [listing DNS information, 7-17](#)
- [listing SNMP community, 7-16](#)
- [listing SP information, 7-16](#)
- [listing system summary information, 7-15](#)
- [listing users, 7-15](#)
- [modifying clock information, 7-22](#)
- [modifying DNS information, 7-21](#)
- [modifying identification information, 7-20](#)
- [modifying Oracle ILOM XML files, 7-2](#)
- [modifying user password, 7-18](#)
- [modifying user role, 7-18](#)
- [overview, 7-1](#)
- [restoring Oracle ILOM defaults, 7-18](#)
- [restoring Oracle ILOM XML files, 7-2](#)
- [SP access over the network, 7-3](#)

**IPv4**

- [listing network settings, 7-16](#)
- [modifying network settings, 7-19](#)

**IPv6**

- [listing network settings, 7-16](#)
- [modifying network settings, 7-20](#)

**L****Linux MMIO access**

[Linux MMIO access settings and Intel network controllers, 5-3](#)

[listing system summary](#)

[ilomconfig, 7-15](#)

[local interconnect, 2-1](#)

**M****manual mode**

- [fwupdate](#)
  - [update subcommand, 5-36](#)

## N

---

- network settings
  - listing IPv4, [7-16](#)
  - listing IPv6, [7-16](#)
  - modifying IPv4, [7-19](#)
  - modifying IPv6, [7-20](#)
- nvmeadm, [8-1](#)
  - error codes, [12-6](#)

## O

---

- Oracle ILOM defaults
  - restoring, [7-18](#)
- Oracle ILOM ID information
  - modifying, [7-20](#)
- Oracle ILOM user
  - creating, [7-18](#)
  - deleting, [7-18](#)
- Oracle ILOM user password
  - modifying, [7-18](#)
- Oracle ILOM user role
  - modifying, [7-18](#)
- Oracle ILOM users
  - listing, [7-15](#)
- Oracle ILOM XML configuration files
  - modifying, [7-2](#)
  - restoring, [7-2](#)
- Oracle Solaris OS
  - biosconfig, [4-4](#)
- overview
  - CLI tools, [1-1](#)

## P

---

- partial disks
  - adding to RAID volume, [9-23](#)
  - creating RAID volumes, [9-22](#)
  - disk display, [9-22](#)
  - exporting RAID configuration, [9-23](#)
  - guidelines for creating RAID, [9-22](#)
  - removing from RAID volume, [9-23](#)

## R

---

- RAID controller configuration
  - clearing, [9-20](#)
  - restoring, [9-20](#)
- RAID volume
  - configuring from a file, [9-21](#)
  - creating, [9-9](#)
  - creating with partial disks, [9-22](#)
  - deleting, [9-11](#)
  - exporting with partial disks, [9-23](#)

- RAID volume (*continued*)
  - name modify, [9-16](#)
- raidconfig, [9-1](#)
  - adding a disk, [9-11](#)
  - adding a spare, [9-12](#)
  - adding partial disks, [9-23](#)
  - auto rebuild disable, [9-15](#)
  - battery backup status, [9-8](#)
  - checking controller configuration, [9-20](#)
  - clearing RAID controller configuration, [9-20](#)
  - command overview, [9-2](#)
  - configuring RAID volumes from a file, [9-21](#)
  - create raid volume, [9-9](#)
  - creating RAID volumes with partial disks, [9-22](#)
  - delete raid volume, [9-11](#)
  - error codes, [12-6](#)
  - export inventory data, [9-21](#)
  - export subcommand, [9-21](#)
  - list subcommand, [9-3](#)
  - modify boot target, [9-15](#)
  - overview, [9-1](#)
  - RAID volume name modify, [9-16](#)
  - removing a disk, [9-12](#)
  - removing a spare, [9-13](#)
  - removing partial disks, [9-23](#)
  - requirements, [9-2](#)
  - restoring RAID controller configuration, [9-20](#)
  - size option
    - disk display, [9-22](#)
    - guidelines, [9-22](#)
    - start task subcommand, [9-17](#)
- restore Oracle ILOM defaults
  - using XML configuration, [7-18](#)

## S

---

- Secure Boot
  - Secure Boot settings and Intel storage devices and controllers, [5-3](#)
- security
  - SP access over the network when using biosconfig, [10-2](#)
  - SP access over the network when using fwupdate, [5-5](#)
  - SP access over the network when using ilomconfig, [7-3](#)
- SNMP community
  - creating, [7-19](#)
  - listing, [7-16](#)
- SP information
  - listing, [7-16](#)
- spare disk
  - adding, [9-12](#)
  - removing, [9-13](#)

SSL certificate checking  
  SP access over the network when using biosconfig, [10-2](#)  
  SP access over the network when using fwupdate, [5-5](#)  
  SP access over the network when using ilomconfig, [7-3](#)

## T

---

TLS remote access encryption  
  SP access over the network when using biosconfig, [10-2](#)  
  SP access over the network when using fwupdate, [5-5](#)  
  SP access over the network when using ilomconfig, [7-3](#)

## U

---

ubiosconfig, [10-1](#)  
  cancel subcommand, [10-11](#)  
  command overview, [10-7](#)  
  error codes, [12-8](#)  
  export subcommand, [10-9](#)  
  exporting UEFI BIOS XML files, [10-1](#)

ubiosconfig (*continued*)  
  import subcommand, [10-10](#)  
  list subcommand, [10-11](#)  
  overview, [10-1](#)  
  reset subcommand, [10-11](#)  
  restoring UEFI BIOS XML files, [10-1](#)  
  SP access over the network, [10-2](#)  
UEFI BIOS configuration files  
  restoring, [10-1](#)  
UEFI BIOS XML configuration files  
  exporting, [10-1](#)

## X

---

XML configuration  
  exporting from Oracle ILOM, [7-11](#)  
  importing to Oracle ILOM, [7-13](#)

## Z

---

zoningcli, [11-1](#)  
  command overview, [11-1](#)  
  disable zoning subcommand, [11-2](#)  
  enable zoning subcommand, [11-2](#)  
  error codes, [12-9](#)  
  list expander subcommand, [11-2](#)