



Server Debug and Provisioning Tool

User Guide

A setup, usage, and troubleshooting guide for Intel server systems.

Rev. 4.2-2

May 2025

Document Revisions

Date	Revision	Changes
October 2020	1.4-1	<ul style="list-style-type: none"> • Python* 3.0 Support • Help modification
November 2020	1.4-2	<ul style="list-style-type: none"> • Support added for these operating systems: <ul style="list-style-type: none"> • Red Hat* Enterprise Linux* 8.x • SUSE* Linux* Enterprise Server 15 • CentOS 8.x and • Ubuntu 20.04 LTS • Python* 3.x support added for these platforms: <ul style="list-style-type: none"> • Intel(R) Server Board S2600 series (Intel(R) Xeon(R) processor E5-2600 v3/v4 product family) • Intel® Server Board (Intel® Xeon® Scalable Processor Family) • Intel® Server Board (2nd Generation Intel® Xeon® Scalable Processor Family) • Intel® Server Board (Intel® Xeon® Platinum 9200 Processor Family) • Custom_deploy update with -no_user_interaction • Update with -no_user_interaction • Get BIOS options with -no_user_interaction • sdptool_update script issue. • Error_Code added for help message
January 2021	1.4-3	<ul style="list-style-type: none"> • Fixed Static scan issues on Intel(R) Server Board S2600 series (Intel(R) Xeon(R) processor E5-2600 v3/v4 product family) • Fixed Vmedia mount failing next time after trying to mount the image using Vmedia with -no_user_interaction switch (even though not supported) • Fixed get SEL when "-" was present in the path to save the file.
March 2021	1.4-4	<ul style="list-style-type: none"> • Redfish Integration to fifteen existing commands (input mechanism will be same as in legacy IPMI as before however the output might slightly change depending upon availability of Redfish on the target machine). • Changes in input and output mechanism for four new commands. • Some changes in the installation scripts to use built-in package managers for installation, update, and removal. • Added new exit error code for invalid length for FRU fields.
March 2021	2.0-0	<ul style="list-style-type: none"> • New Platform M70KLP support added. • Storageinfo and nicinfo added, which will provide information regarding storage and network devices.
April 2021	2.0-1	<ul style="list-style-type: none"> • Added support for online firmware version checker • Minor updates throughout for clarity
June 2021	2.1-1	<ul style="list-style-type: none"> • Added support for online firmware version checker API based • SEL command supporting all severity combinations • Added -ipmi command to force Intel® Server Debug and Provisioning Tool to use Legacy / IPMI interface.
August 2021	2.1-2	<ul style="list-style-type: none"> • Added support for Intel® Server Board D40AMP
September 2021	2.2-0	<ul style="list-style-type: none"> • Fixed /usr/local/SDPTool directory accessible by non-root user • Fixed health status in sensor output via Redfish • Added "EventID" and "Sensor Number" fields in Redfish SEL output • Added Firmware Version and Name of RAID Controllers for storageinfo command • Added Firmware Version for nicinfo command
November 2021	3.0-0	<ul style="list-style-type: none"> • Added support for Intel® Server Board M20NTP (North Pass) • End of life of Python2 Supported distros • Fixed storageinfo command • Fixed update issue on older BMC • Added CPLD version for softwareinventory

Date	Revision	Changes
February 2022	4.0-0	<ul style="list-style-type: none"> Added predefined configurations support for these commands using JSON configuration file: <ul style="list-style-type: none"> vmedia update custom_deploy set_biosconfig_all Removed XML configuration file support from update command Added -clean option for unmount
March 2022	4.1-0	<ul style="list-style-type: none"> Added cup_deploy command
June 2022	4.1-1	<ul style="list-style-type: none"> Fixed BIOS recovery update failure on Intel® Server Platforms M50CYP and D50TNP.
July 2022	4.1-2	<ul style="list-style-type: none"> Signed RPM and DEB files
October 2022	4.2-0	<ul style="list-style-type: none"> Added -staged, -attached and -detached flag support for cup_deploy command Added support for cup_inspect command Added config file support Support added for below operation system <ul style="list-style-type: none"> Red Hat* Enterprise Linux* 9.x Ubuntu 22.04 LTS
February 2023	4.2-1	<ul style="list-style-type: none"> Added support for Intel® Server M50FCP Added support for Intel® Server D50DNP Added JSON config file support for cup_inspect
May 2025	4.2-2	<ul style="list-style-type: none"> From the Eagle stream generation, remove supported command softwareinventory, setlan, disablelan, setlanipv6, disablelanipv6, failover, nodeposition, setlandhcp, setlandhcpipv6, setlanicmpv6, kvm, custom_deploy. Remove VMCLI

Disclaimers

MiTAC/Intel's features and benefits depend on system configuration and may require enabled hardware, software, or service activation. Performance varies depending on system configuration. No computer system can be absolutely secure. Check with your system manufacturer or retailer or learn more at intel.com.

You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning MiTAC/Intel products described herein. You agree to grant MiTAC/Intel a non-exclusive, royalty-free license to any patent claim thereafter drafted which includes subject matter disclosed herein.

No license (express or implied, by estoppel or otherwise) to any intellectual property rights is granted by this document.

The products described may contain design defects or errors known as errata, which may cause the product to deviate from published specifications. Current characterized errata are available on request.

MiTAC/Intel disclaims all express and implied warranties, including without limitation, the implied warranties of merchantability, fitness

for a particular purpose, and non-infringement, as well as any warranty arising from course of performance, course of dealing, or usage in trade.

MiTAC logo are trademarks of MiTAC International Corporation or its subsidiaries/affiliates in Taiwan and/or other countries.

Intel and the Intel logo are trademarks of Intel Corporation or its subsidiaries in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

Copyright © MiTAC Computing Technology Corporation/Intel Corporation. All rights reserved.

MiTAC Computing Technology Corporation,

3F., No. 1, R&D Road 2, Hsinchu Science Park, Hsinchu 30076, Taiwan, R.O.C

Intel Corporation, 2200 Mission College Blvd.,
Santa Clara, CA 95052-8119, USA.

Table of Contents

Chapter 1. Introduction	10
1.1 Document Scope	10
1.2 System Requirements	10
1.3 Terminology	11
1.4 Related Documents	11
1.5 Support for Intel Products	11
Chapter 2. Get Started	12
2.1 Prerequisites for Installation	12
2.1.1 Prerequisite Packages	12
2.2 Installation Steps.....	12
2.3 Removal Steps.....	13
2.4 Update Steps	13
2.5 Intel® Data Center Manager (Intel® DCM) integration.....	13
2.5.1 SDPTool Configuration File	13
2.5.2 Update	17
2.5.3 Adding a Managed Server for a Custom Deploy.....	18
2.5.4 Sudoers	19
2.6 User Privileges.....	19
2.7 Network Ports.....	19
Chapter 3. Feature Script	20
3.1 General Rules.....	20
3.2 Global Arguments	20
3.3 Update Firmware.....	21
3.3.1 SUP Folder Based	21
3.3.2 Config File Based.....	21
3.3.3 Config File Based custom_deploy.....	21
3.3.4 Custom Deploy.....	21
3.3.5 Custom Folder Based	22
3.3.6 Configuration File Based	22
3.4 Set Options.....	22
3.5 Set BIOS Config (Redfish* version of setoptions).....	22
3.6 Deploy Options	23
3.7 Set BIOS Config All (Redfish* version of deployoptions)	23
3.7.1 Configuration File Based	23
3.8 Get BIOS Options	23
3.9 Get BIOS Config (Redfish* version of getbiosoptions)	24
3.10 Get INI.....	24
3.11 Get BIOS Config All (Redfish* Version of getini).....	24
3.12 KVM.....	24

3.13	Vmedia.....	25
3.13.1	IPMI Based.....	25
3.13.2	Redfish* Based	25
3.13.3	Config File Based.....	25
3.14	IPMI	26
3.15	Power	26
3.16	Sensor.....	26
3.17	SEL.....	26
3.18	Set LAN.....	26
3.19	Disable LAN	27
3.20	Set LAN IPV6	28
3.21	Disable LAN IPV6.....	28
3.22	LAN Fail Over	28
3.23	Node Position	29
3.24	System Information.....	29
3.25	FRU	29
3.26	Memory Information.....	29
3.27	CPU Information.....	29
3.28	Memory Temperature	29
3.29	Power Statistics.....	30
3.30	Set LAN DHCP.....	30
3.31	Set LAN DHCP IPV6.....	30
3.32	Set LAN Stateless ICMPV6	30
3.33	Debug Log	30
3.34	Supported Updates.....	31
3.35	Unmount.....	31
3.35.1	IPMI Based.....	31
3.35.2	Redfish* Based	31
3.36	Storage Information.....	31
3.37	NIC Information.....	31
3.38	Online Firmware (BIOS/BMC/ME/FRUSDR/DCPMM/SSD) Version Check	32
3.39	Custom Update Package Inspect.....	33
3.39.1	Config File Based.....	34
3.40	Custom Update Package Deploy.....	34
3.40.1	CUP Deploy modes.....	34
3.40.2	Config File Based.....	35
Chapter 4.	custom_deploy.....	36
Chapter 5.	SDP Tool Configuration file.....	37
Chapter 6.	CUP	38
Chapter 7.	Troubleshooting Tips	39
7.1	SSH Command "sudo: you must have a tty to run sudo" Error.....	39

7.2	Tar Time Stamp Messages	39
7.3	Java* Version Required (Java* 1.7)	40
7.4	Proxy Settings.....	40
Chapter 8.	Error Codes	41
Chapter 9.	IPMI Reboot Features List.....	44
Chapter 10.	Redfish* Features List.....	45
Chapter 11.	Appendix A. Unsupported command list.....	47
Chapter 12.	Appendix B. Unsupported command list in IPMI based	48
Chapter 13.	Appendix C. Removed package	49

List of Tables

Table 1. Operating Systems and Intel® Server Products Supported..... 10

Table 2. Terminology..... 11

Table 3. Configuration File – Key Definitions 16

Table 4. Error Codes 41

Table 5. IPMI Reboot Features List 44

Table 6. Redfish* Features List 45

Table 7. Unsupported command list 47

Table 8. Unsupported command in IPMI based..... 48

List of Figures

Figure 1. Proxy Setting Configuration.....	13
Figure 2. FRU Print Details.....	15
Figure 3. Server Tool Search Results	38
Figure 4. CUP Download	38
Figure 5. SSH Command Error.....	39
Figure 6. Tar Time Stamp Messages	39
Figure 7. Java* Version	40

Chapter 1. Introduction

The Server Debug and Provisioning Tool (SDP Tool) is a tool that runs from a single server to debug and provision multiple Intel server boards and systems through the out of band BMC interface.

The SDP Tool is designed to work with the following Intel server product families:

- Intel® Server S2600WT/S2600WTR Family
- Intel® Server S2600KP/S2600KPR Family
- Intel® Server S2600TP/S2600TPR Family
- Intel® Server S2600CW/S2600CWR Family
- Intel® Server S2600WF/S2600WFR Family
- Intel® Server S2600ST/S2600STR Family
- Intel® Server S2600BP/S2600BPR Family
- Intel® Server S9200WK Family
- Intel® Server M70KLP Family
- Intel® Server D50TNP Family
- Intel® Server M50CYP Family
- Intel® Server D40AMP Family
- Intel® Server M20NTP Family
- Intel® Server M50FCP Family
- Intel® Server D50DNP Family

1.1 Document Scope

The purpose of this user guide is to help system/server administrators install and use the Server Debug and Provisioning Tool (SDP Tool). This guide provides information on the features and benefits of SDP Tool, software requirements, and the supported operating systems and platforms. This guide also explains the installation and removal process.

Note: Refer to the *Server Debug and Provisioning Tool Release Notes* for known issues on platforms and during the installation.

1.2 System Requirements

Table 1. Operating Systems and Intel® Server Products Supported

Intel® Server Products	Operating System Version
<ul style="list-style-type: none"> • Intel® Server S2600WT/S2600WTR Family • Intel® Server S2600KP/S2600KPR Family • Intel® Server S2600TP/S2600TPR Family • Intel® Server S2600CW/S2600CWR Family • Intel® Server S2600WF/S2600WFR Family • Intel® Server S2600ST/S2600STR Family • Intel® Server S2600BP/S2600BPR Family • Intel® Server S9200WK Family • Intel® Server D50TNP Family • Intel® Server M50CYP Family • Intel® Server M70KLP Family • Intel® Server D40AMP Family • Intel® Server M20NTP Family • Intel® Server M50FCP Family • Intel® Server D50DNP Family 	<ul style="list-style-type: none"> • Red Hat Enterprise Linux* 8.x 64 Bit and 9.x 64 Bit. • SUSE Linux Enterprise Server* 15 64 Bit • Ubuntu 20.04 LTS and Ubuntu 22.04 LTS

1.3 Terminology

Table 2. Terminology

Term	Definition
BMC	Baseboard management controller. A discrete server within the server system that operates independently of the operating system and allows you to manage a system remotely, even in the absence of the operating system.
CLI	Command-line interface
FRU	Field replaceable unit
IPMI	Intelligent Platform Management Interface. A command protocol for interacting with the BMC over the network or from the local system.
Redfish*	Operates over HTTPs using a REST API independently of the operating system and allows management of a system remotely using basic https commands like post, get, patch.
LAN	Local area network
Management Server	Server system where the SDP Tool is installed. It acts as a host server which has network connectivity to the rest of the managed servers.
Managed Server	Intel server system in a cluster or data center that is accessible and managed by the management server.
OS	Operating system
Out-of-band	Refers to management of a system other than in-band (by executing applications within the system's installed operating system)
SDR	Sensor data record
SEL	System event log
sudo	"super user do". This is Linux command that permits programs to be executed as a super user (also called <i>root user</i>) or another user. In Windows, the equivalent is the runas command.
SUP	System update package

1.4 Related Documents

IPMI-Intelligent Platform Management Interface Specification, 2nd Generation, v2.0 (available here: <https://www.intel.com.tw/content/www/tw/zh/products/docs/servers/ipmi/ipmi-second-gen-interface-spec-v2-rev1-1.html>)

1.5 Support for Intel Products

Visit <https://www.intel.com/content/www/us/en/support.html>

for current technical support information and updates.

Chapter 2. Get Started

2.1 Prerequisites for Installation

The following tools must be installed before the installation of the SDP Tool to ensure proper functionality. The SDP Tool is an RPM-based package and fails to install if the following prerequisites are not met. These prerequisites may be included with the standard distribution CD/DVD or .iso, or may be installed from a configured repository.

2.1.1 Prerequisite Packages

The following application packages are required for installing the SDP Tool:

- Python* 3.x (Distro in-built only)
- ipmitool* 1.8.18
- curl* 7.29.0
- OpenSSL* 1.0.0x higher
- wget* 1.16 higher
- Python-requests
- OpenIPMI* drivers

2.2 Installation Steps

Install all the listed prerequisites to the Management Server and verify that they are accessible.

Install the SDP Tool on the Management Server by using the following steps:

1. Download or copy the SDP Tool package, **SDPTool-x.y-z.tar.gz**, to the target directory.
2. Untar the **tar.gz** file:

```
Prompt #> tar -xvzmf SDPTool-x.y-z.tar.gz
```

3. Go to untarred SDPTool-x.y-z directory folder. Run `sdptool_install.sh` to install the package. For example:

```
Prompt #> cd <path/to/SDPTool-x.y-z>
```

```
Prompt #> ./sdptool_install.sh
```

If an older version is present, uninstall it first by using the command:

```
Prompt #> ./sdptool_uninstall.sh
```

Or use the update script:

```
Prompt #> ./sdptool_update.sh
```

4. If necessary for the network environment where the Management Server is installed, Configure proper proxy settings for your environment.

Notes:

- Administrator privileges are needed to install / uninstall the utility.
 - Proxy may be required for getting the packages from pip.
 - Proxy setting may also be needed to set as environment variable. Contact your system/network administrator for more details. Check [Section 7.4](#) for FAQ's.
-

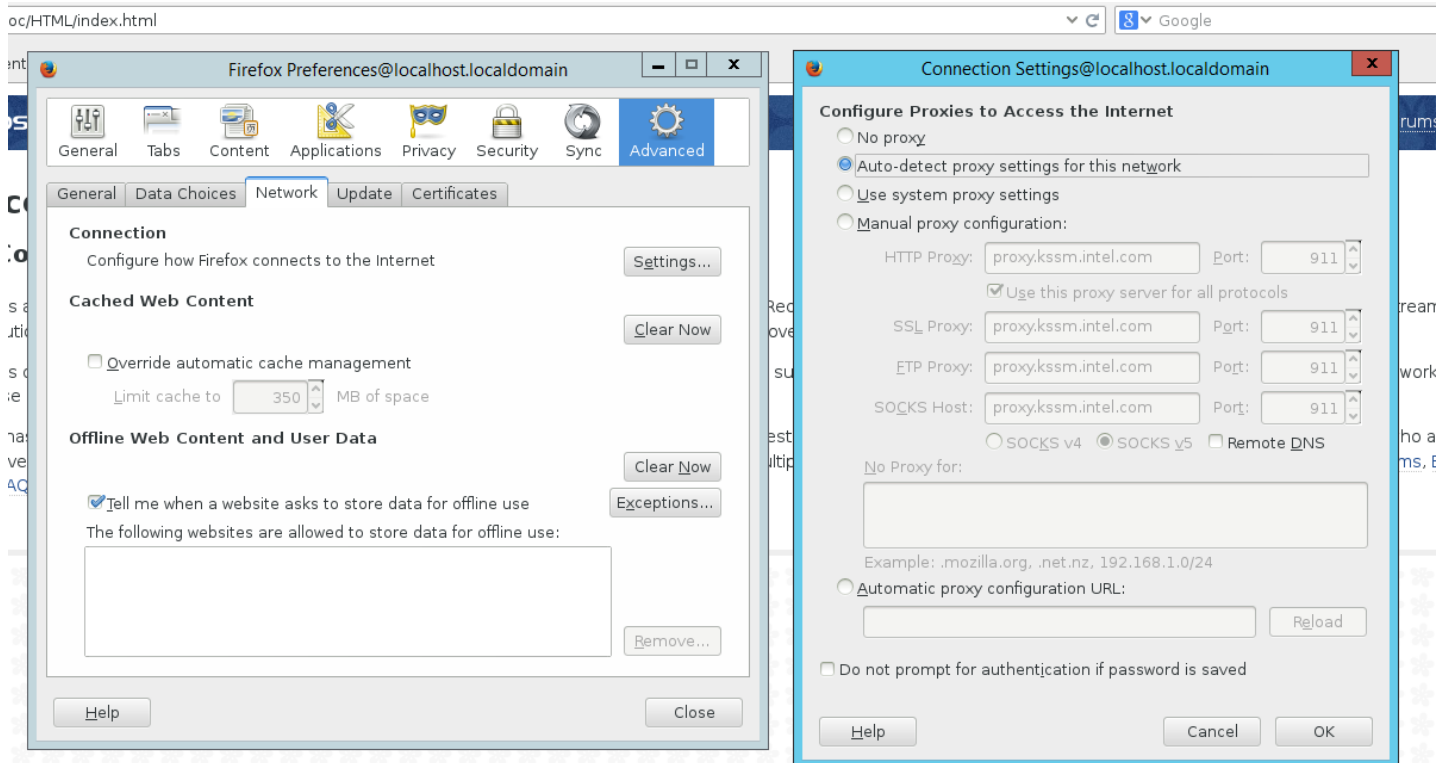


Figure 1. Proxy Setting Configuration

2.3 Removal Steps

To uninstall the package, enter the following commands:

Navigate to the extracted SDPTool directory. If removed, extract the SDPTool directory again.

```
Prompt #> tar -xvzmf SDPTool-x.y-z.tar.gz
```

```
Prompt #> cd <path/to/sdptool-x.y-z>
```

```
Prompt #> ./sdptool_uninstall.sh
```

2.4 Update Steps

To update the package, enter the following commands:

```
Prompt #> tar -xvzmf SDPTool-x.y-z.tar.gz
```

```
Prompt #> cd <path/to/sdptool-x.y-z>
```

```
Prompt #> ./sdptool_update.sh
```

2.5 Intel® Data Center Manager (Intel® DCM) integration

[Intel® Data Center Manager](#) (Intel® DCM) is a management solution stack. It provides accurate real-time monitoring (thermal and power), management, and platform update capabilities (e.g., BIOS and BMC) on Intel server boards and systems. For more information and a feature list of Intel® DCM, refer to the user guide that comes with the download of the tool. SDP Tool supports Intel® DCM by providing the following capabilities from a JSON-based configuration file.

2.5.1 SDPTool Configuration File

The SDP Tool configuration file is a user-written JSON file. The JSON supports the following commands:

1. update
2. custom_deploy

3. vmedia
4. set_biosconfig_all
5. cup_deploy
6. cup_inspect

The following sample configuration file passed to the SDP Tool provides the locations of various SUP and custom folders that a user can provide instead of providing them in the command line.

```
{
  "Version": "1.0",
  "FRUDeviceDescription": "RackMount",
  "FruField": "Model",
  "Products": {
    "S2600WFT": {
      "vmedia": {
        "image": "nfs://<NFS_IP>/<PATH_TO_ISO>"
      },
      "set_biosconfig_all": {
        "iniFile": "<PATH_TO_SYSCFG.INI>"
      },
      "cup_deploy": {
        "cup": "<PATH_TO_CUP.zip>",
        "staged": false
      },
      "cup_inspect": {
        "cup": "<PATH_TO_CUP.zip>",
        "format": "<PATH_TO_FILE.json>"
      },
      "update": {
        "SUPFolder": "<PATH_TO_SUP>",
        "CustomUpdate": {
          "CustomFolder": "<PATH_TO_CUSTOM_DEPLOY_DIR>"
        }
      },
      "custom_deploy": {
        "CustomFolder": "<PATH_TO_CUSTOM_DEPLOY_DIR>",
        "Args": "-custom_deploy_args"
      }
    }
  }
}
```

Notes:

- SUP is a managed Server Update Package provided by Intel. The SUP is distributed as a .zip file. Uncompressing the .zip file yields the SUP folder. The SUP package contains the required scripts and images/binaries that run in the EFI Shell to perform the necessary updates of the Managed Server.
 - The Custom Folder is a user-defined folder. This allows users to write their own scripts that run in the EFI Shell on the managed server. Refer to [Chapter 4](#) for more details.
-

1. The key names are case sensitive.
2. FRUDeviceDescription and FruField come from the FRU information seen in the following image.

```

FRU Device Description : RackMount
  Manufacturer:      Intel Corporation
  Model:             S2600WFT
  Serial Number:     .....
  Part Number:       .....
FRU Device Description : RackMount/Baseboard
  Manufacturer:      Intel Corporation
  Model:             S2600WFT
  Serial Number:     BQWF83600185
  Part Number:       H48104-863
FRU Device Description : RackMount/FrontPanel
  Manufacturer:      Intel Corporation
  Model:             FFPANEL
  Serial Number:     BQWL83403135
  Part Number:       H39380-171
FRU Device Description : RackMount/HSBackplane1
  Manufacturer:      Intel Corporation
  Model:             FR1304S3
  Serial Number:     BQWL83501065
  Part Number:       G97162-371
FRU Device Description : RackMount/PCIEriser1
  Manufacturer:      Intel Corporation
  Model:             F1UL16RISER3
  Serial Number:     BQWK83501666
  Part Number:       H88399-250
FRU Device Description : RackMount/PCIEriser2
  Manufacturer:      Intel Corporation
  Model:             F1UL16RISER3
  Serial Number:     BQWK83503313
  Part Number:       H88399-250
FRU Device Description : RackMount/PwrSupply1FRU
  Manufacturer:      FLEXTRONICS
  Model:             S-1100ADU00-201
  Serial Number:     EXWD83502202
  Part Number:       G84027-009

```

Figure 2. FRU Print Details

Table 3. Configuration File – Key Definitions

Key	Description	Sample Value
Version	Version of the JSON used	1.0
FRUDeviceDescription	FRU Device Description section that is seen in the SDPTool fru print. Refer to Section 3.25 .	Rackmount
FruField	Field to be used in the identified by the FRUDeviceDescription value	Model
Products	Main section containing all the pre-defined values	-
vmedia	Pre-definition for the SDPTool vmedia Command. Refer to Section 3.13.3	-
image	The key indication the file to use for virtual mount	"vmedia": { "image": "nfs://<NFS_IP>/<PATH_TO_ISO>" }
set_biosconfig_all	Pre-definition for the SDPTool set_biosconfig_all Command. Refer to Section 3.7.1	-
iniFile	The key indication the file to use for BISO Configurations	"set_biosconfig_all": { "iniFile": "<PATH_TO_SYSCFG.INI>" }
cup_deploy	Pre-definition for the SDPTool cup_deploy Command. Refer to Section 3.40	-
cup	The key indication the cup package to use for cup deploy	"cup_deploy": { "cup": "<PATH_TO_CUP.zip>", "staged": true }
cup_inspect	Pre-definition for the SDPTool cup_inspect Command. Refer to Section 3.39	-
update	Pre-definition for the SDPTool update Command. Refer to Section 3.3.2	-
SUPFolder	Path to the SUP folder, which has the firmware for Managed Server update	"SUPFolder": "<PATH_TO_SUP>"
CustomUpdate	Pre-definition for the SDPTool Custom_Deploy Command. Refer to Section 3.3.3	"CustomUpdate": { "CustomFolder": "<PATH_TO_CUSTOM_DEPLOY_DIR>", "CustomDeployArgs": "-custom_deploy_args" }
CustomFolder	Path to the custom folder, which has the custom scripts	"CustomUpdate": { "CustomFolder": "<PATH_TO_CUSTOM_DEPLOY_DIR>", "CustomDeployArgs": "-custom_deploy_args" }
CustomDeployArgs	Arguments for custom_deploy.nsh	"CustomUpdate": { "CustomFolder": "<PATH_TO_CUSTOM_DEPLOY_DIR>", "CustomDeployArgs": "-custom_deploy_args" }
custom_deploy	Pre-definition for the SDPTool Custom_Deploy Command. Refer to Section 3.3.6	"custom_deploy": { "CustomFolder": "<PATH_TO_CUSTOM_DEPLOY_DIR>", "CustomDeployArgs": "-custom_deploy_args" }

2.5.2 Update

To add different managed servers for an SUP-based update, add the highlighted content and make the necessary changes to the values:

```
{
  "Version" : "1.0",
  "FRUDeviceDescription": "RackMount",
  "FruField": "Model",
  "Products" : {
    "S2600WFT" : {
      "vmedia": {
        "image": "nfs://<NFS_IP>/<PATH_TO_ISO>"
      },
      "set_biosconfig_all": {
        "iniFile": "<PATH_TO_SYSCFG.INI>"
      },
      "cup_deploy": {
        "cup": "<PATH_TO_CUP.zip>",
        "staged": true
      },
      "cup_inspect": {
        "cup": "<PATH_TO_CUP.zip>",
        "format": "<PATH_TO_FILE.json>"
      },
      "update": {
        "SUPFolder": "<PATH_TO_SUP>",
        "CustomUpdate": {
          "CustomFolder": "<PATH_TO_CUSTOM_DEPLOY_DIR>",
          "Args": "-custom_deploy_args"
        }
      }
    },
    "custom_deploy": {
      "CustomFolder": "<PATH_TO_CUSTOM_DEPLOY_DIR>",
      "Args": "-custom_deploy_args"
    }
  }
}
```

Note: "S2600WFT" corresponds to the Board Product value in the FRU field of the Managed Server. The tags are case sensitive.

2.5.3 Adding a Managed Server for a Custom Deploy

To add a different managed server for the custom update feature, add the highlighted content and make the necessary changes to the values:

```
{
  "Version": "1.0",
  "FRUDeviceDescription": "RackMount",
  "FruField": "Model",
  "Products": {
    "S2600WFT": {
      "vmedia": {
        "image": "nfs://<NFS_IP>/<PATH_TO_ISO>"
      },
      "set_biosconfig_all": {
        "iniFile": "<PATH_TO_SYSCFG.INI>"
      },
      "cup_deploy": {
        "cup": "<PATH_TO_CUP.zip>",
        "staged": true
      },
      "cup_inspect": {
        "cup": "<PATH_TO_CUP.zip>",
        "format": "<PATH_TO_FILE.json>"
      },
      "update": {
        "SUPFolder": "<PATH_TO_SUP>",
        "CustomUpdate": {
          "CustomFolder": "<PATH_TO_CUSTOM_DEPLOY_DIR>"
        }
      },
      "custom_deploy": {
        "CustomFolder": "<PATH_TO_CUSTOM_DEPLOY_DIR>",
        "Args": "-custom_deploy_args"
      }
    }
  }
}
```

Note: "S2600WFT" is the Model value in the FRU field of the Managed Server.

1. The tags are case sensitive.
2. The value for the FruField can be obtained from one of the left columns of the FRU print output. Corresponding values in the right column form the next tag. In the previous example, the FRU_Field is a "Model", with the value being "S2600WFT". The FRU fields can be obtained by using the command in [Section 3.25](#), refer to [Figure 2](#).

2.5.4 Sudoers

Add the DCM user to the sudoers list. SDP Tool works only as the root or if the user is part of the sudoers list.

```
#> visudo
```

Add the following line in the file:

```
<DCM_USERNAME> ALL=(ALL) /usr/bin/SDPTool
```

2.6 User Privileges

The IPMI interface is used for commands. The SDP Tool requires the user to have administrative privileges, otherwise the commands return an error with a message to check the user's credentials and privileges.

2.7 Network Ports

The following network ports are used by the SDP Tool to connect to the Managed Server:

1. Ping: No TCP port is used, but ICMP packets need to be allowed.
2. https: Server port outbound '**443**' and '**916**'.
3. http: Server port outbound '**80**'.
4. ipmi: Server port outbound '**623**' and '**627**'.
5. kvm: Server port '**5902**' and '**7578**'.
6. vmedia: Server port '**5123**'.

Note: The SDP Tool may use a combination of these ports to complete an operation.

Chapter 3. Feature Script

The SDP Tool script is the main engine of SDP Tool OOB features. This section explains the methods to execute SDP Tool features and the objectives accomplished by executing them.

3.1 General Rules

To display the usage menu, enter -h.

Example: SDPTool -h

Each valid operation run creates logs in ./Logfiles/<target_ip>/<operation>.log

For example, after running SDPTool 192.168.1.10 bmcuser bmcpw powerstatistics, there are logs in the location:

```
/usr/local/log/SDPTool/Logfiles/192_168_1_10/powerstatistics.log
```

Any failure generates an *.err log file in the logfiles directory (/usr/local/log/SDPTool/Logfiles/<IP>/<operation>.err). The specific definition of each error code is available in [Chapter 8](#).

For features that invoke a Managed Server reboot, only one operation for an IP may be run at any given time. Any attempt to execute another operation that invokes a reboot on the same IP responds in an undefined way, potentially interfering with the current operation being performed on the Managed Server having that IP address. The SDP Tool restricts the user from running two different operations that result in rebooting the same Managed Server.

For example:

```
SDPTool 192.168.1.10 bmcuser bmcpw getini
SDPTool 192.168.1.10 bmcuser bmcpw set_biosconfig 'QuietBoot 1'
SDPTool 192.168.1.10 bmcuser bmcpw custom_deploy customdeployfolder/
```

Notes:

- These operations must not be run at the same time, as all of them reboot the system. Meaning that these operations would interfere with each other's operation and reboot the managed system.
 - Some of the commands that reboot the Managed Server require the Managed Server to be in the KCS policy mode **Allow All** to work. The tool errors out if the KCS policy mode is Restricted or Deny All.
 - Refer to the list of commands that perform the reboot of the Managed Server in [Chapter 9](#). Additionally, users can find the commands that would reboot the Managed Server by running prompt #> SDPTool -h. This informs the user if the command will reboot the target Managed Server.
 - The SDP Tool suite is not supported when the Secure Boot option is enabled in BIOS configuration.
-

3.2 Global Arguments

The user can provide additional arguments that are common for all the commands. These arguments are the following:

- -no_user_interaction: No prompting for any action, the default values are chosen
- -softreset: Begin by performing a soft reset on the system in case the system is in OS mode.
- -no_reboot: Certain commands require a system reset to take effect. This option does not perform a reset of the system and the user must explicitly invoke a reboot through another method for changes to take effect.
- -ipmi: Force SDP Tool to use legacy/IPMI interface.

3.3 Update Firmware

There are two variants of the update firmware now available via IPMI and two variants via Redfish*.

3.3.1 SUP Folder Based

```
SDPTool <ipv4> <username> <password> update <SUP_folder> [-no_user_interaction] [-softreset]
```

To update the BIOS/ME/BMC/SDR system firmware, a SUP package must be used instead of a FSUP package. This feature uses flash utilities and images within the SUP package.

Example: SDPTool 192.168.1.10 admin admin123 update SUP/S2600WT

Notes:

- SUP_Folder – Path to the system update package (SUP) is required and must be provided as an argument.
 - SDPTool does not support BMC recovery region update via redfish.
 - Not supported argument -softreset from Eagle Stream generation onwards.
-

3.3.2 Config File Based

```
SDPTool <ipv4> <username> <password> update -c <config file> [-no_user_interaction] [-softreset]
```

This option provides the user the ability to provide a configuration file instead of the SUP folder path. The creation of the configuration file is covered in [Section 2.5](#). The configuration file is used to provide the necessary information, such as SUP folder, for the Managed Server being updated.

The other options are same as in [Section 3.3.1](#).

Note: To verify the update, check the versions of the firmware. The versions can be checked using the systeminfo command. Refer to [Section 3.24](#) for more details. Not supported argument -softreset from Eagle Stream generation onwards.

3.3.3 Config File Based custom_deploy

```
SDPTool <ipv4> <username> <password> update -c <config file> [-no_user_interaction] [-softreset] --all
```

This option provides the user the ability to provide a configuration file instead of the custom_folder path. The creation of the configuration file is covered in the [Section 2.5](#). The configuration file is used to provide the necessary information, such as the correct custom folder, for the Managed Server.

--all: Use this option to perform a custom update. The custom_folder details are taken from the configuration file provided as the command argument to perform the custom update.

Note: The command used is update, not custom_deploy, unlike in [Section 3.3.5](#). The --all option is necessary to perform the custom update, otherwise the update command performs a SUP-based Managed Server update. Not supported argument -softreset from Eagle Stream generation onwards.

3.3.4 Custom Deploy

The custom deploy feature is update-based, using a user-defined folder to perform the action that the user desires instead of using an SUP provided by Intel. Custom deploy can be used to perform various custom updates, which include but are not limited to firmware updates for OEM parts (like SSD, NIC, HBA).

[Chapter 4](#) contains further information detailing the creation of a custom deployment folder and the scripts associated with the custom deployment folder.

3.3.5 Custom Folder Based

```
SDPTool <ipv4> <username> <password> custom_deploy <folder name which contains deploy.nsh>  
<"argument(s) for deploy.nsh"> [-no_user_interaction] [-softreset]
```

To deploy a user customized script, the customized script must be in or launched from the deploy.nsh script.

Note: A reboot is required for this option, clearing the EFI mailbox.

- `deploy_result.log`: The output from `deploy.nsh` can be redirected to this file name. The file is saved to the `Logfiles/ip` folder and the content is displayed to the terminal after `custom_deploy` script with extra argument(s) is executed.
- `deploy_details.log`: The details from `deploy.nsh` can be redirected to this file name. The file will be saved to the `Logfiles/ip` folder after `custom_deploy` script with extra argument(s) being executed.

Example: `SDPTool 192.168.1.10 admin admin123 custom_deploy folder_with_nsh_file`

Example: `SDPTool 192.168.1.10 admin admin123 custom_deploy folder_with_nsh_file "argument1 argument2 argument3"`

3.3.6 Configuration File Based

```
SDPTool <ipv4> <username> <password> custom_deploy -c <config file> [-no_user_interaction] [-softreset]
```

This command provides the user the ability to provide a configuration file instead of the `custom_folder` path. The creation of the configuration file is covered in the [Section 2.5](#). The configuration file is used to provide the necessary information, such as the correct custom folder, for the Managed Server.

3.4 Set Options

```
SDPTool <ipv4> <username> <password> setoptions <"syscfg arguments"> [-no_user_interaction] [-softreset]
```

This option configures BIOS/BMC settings by executing `syscfg` command-line arguments. Refer to the *Server Configuration Utility User Guide* for specific `syscfg` command-line arguments.

Note: A reboot is required for this option.

Example: `SDPTool 192.168.1.10 admin admin123 setoptions /i`

3.5 Set BIOS Config (Redfish* version of setoptions)

```
SDPTool <ipv4> <username> <password> set_biosconfig <"var new_val"> [-no_user_interaction] [-no_reboot]
```

This option configures BIOS/BMC settings via Redfish* API. Please refer to the file (`syscfg.INI`) content obtained through the command `"get_biosconfig_all"` to set `var` and `new_val`. Reference the following step.

1. Get system bios configuration file by command `get_bioconfig_all`.
Example : `192.168.1.10 admin admin123 get_biosconfig_all`
After execute this command the file `syscfg.INI` will be generated in default SDPTool log file location.
2. Open file `syscfg.INI` to know what bios option could be set.
Choose any bios option you want to update, remember you must be type exactly the same, include uppercase/lowercase. Please note different server may have different settings. For example your probably get `'QuietBoot = 1'` in one server but get `'QuietBoot = Enabled'` in another server.

Note: A reboot is required for this option. If the `-no_reboot` argument is used, the BIOS and Intel® ME updates will not be applied until the Managed Server is rebooted manually later by the user.

Example: `SDPTool 192.168.1.10 admin admin123 set_biosconfig 'QuietBoot 1'`

3.6 Deploy Options

`SDPTool <ipv4> <username> <password> deployoptions <restore filename> [-no_user_interaction] [-softreset]`

This option configures BIOS/BMC settings using the `syscfg ini` method. Once the `.ini` file is provided, the user may change many of the BIOS options and set them within one command.

Note: The `.ini` provided for deploy options must have valid values. A reboot is required for this option, clearing the EFI mailbox.

Example: `SDPTool 192.168.1.10 admin admin123 deployoptions sysconfig.ini`

3.7 Set BIOS Config All (Redfish* version of deployoptions)

`SDPTool <ipv4> <username> <password> set_biosconfig_all <restore filename> [-no_user_interaction] [-no_reboot]`

This option configures BIOS/BMC settings using a `.ini` file via the Redfish* method. A `.ini` file can be generated using the `SDPTool "get_biosconfig_all"` command. Once `.ini` file is generated, the user may change one or more BIOS configuration options and set them within one command.

Note: A reboot is required for this option.

Example: `SDPTool 192.168.1.10 admin admin123 set_biosconfig_all sysconfig.ini`

3.7.1 Configuration File Based

`SDPTool <ipv4> <username> <password> set_biosconfig_all -c <config file> [-no_user_interaction] [-no_reboot]`

A configuration file can also be provided to take the defaults paths and files required for performing the action. The configuration file is used to pick the pre-defined `ini` file for the set operation.

3.8 Get BIOS Options

`SDPTool <ipv4> <username> <password> getbiosoptions <"option to retrieve"> [-no_user_interaction] [-softreset]`

This option returns the value of a specific BIOS setting that is supported by the Server Configuration Utility. Refer to the *Server Configuration Utility User Guide* for specific `syscfg` command-line arguments.

Please refer to the file content obtained through the command `"getini"` to set option. Reference the following step.

1. Get bios options by command `getini`
Example `SDPTool 192.168.1.10 admin admin123 getini`
After execute this command file `getini_syscfg.INI` will be generated in default `SDPTool` log file location.
2. Open file `getini_syscfg.INI` to know what bios option could be set.

Choose any bios option you want to know. You must be type exactly the same, include uppercase / lowercase and white space. Please note the file getini_syscfg.INI and syscfg.INI are similar but different. For example you could find bios options
"Quiet Boot=Enabled ;Options: Disabled=00: Enabled=01",
If you want to know Quiet Boot current value, type "Quiet Boot" as option to retrieve.

Note: A reboot is required for this option, clearing the EFI mailbox.

Example: SDPTool 192.168.1.10 admin admin123 getbiosoptions "Quiet Boot"

3.9 Get BIOS Config (Redfish* version of getbiosoptions)

SDPTool <ipv4> <username> <password> get_biosconfig <"option to retrieve">

This option returns the value of a specific BIOS setting via Redfish* API.

Note: A reboot is not required.

Example: SDPTool 192.168.1.10 admin admin123 get_biosconfig "QuietBoot"

3.10 Get INI

SDPTool <ipv4> <username> <password> getini [-no_user_interaction] [-softreset]

This command returns BIOS/BMC settings using the syscfg /save .ini file method.

Note: A reboot is required for this option, clearing the EFI mailbox.

Example: SDPTool 192.168.1.10 admin admin123 getini

3.11 Get BIOS Config All (Redfish* Version of getini)

SDPTool <ipv4> <username> <password> get_biosconfig_all

This command returns BIOS/BMC settings using the Redfish* API.

Note: A reboot is not required.

Example: SDPTool 192.168.1.10 admin admin123 get_biosconfig_all

3.12 KVM

Deprecated: *This is a Deprecated command, and no further support will be provided*

SDPTool <ipv4> <username> <password> kvm launch

This command launches KVM windows for remote control.

Example: SDPTool 192.168.1.10 admin admin123 kvm launch

Note: Not supported from Eagle Stream generation onwards.

3.13 Vmedia

3.13.1 IPMI Based

```
SDPTool <ipv4> <username> <password> vmedia <IMAGE/ISO> [-no_user_interaction]
```

The vmedia command allows the addition of virtual media in .img/.iso format to the remote machine. Add the relevant virtual media by redirecting the image/ISO file specified. Up to 4 .img/.iso are allowed to be mounted in parallel.

Note: Mounting the same image again on the same target system is not allowed. Not supported in IPMI based from Eagle Stream generation onwards.

Examples:

```
SDPTool 192.168.1.10 admin admin123 vmedia image.img
```

```
SDPTool 192.168.1.10 admin admin123 vmedia image.iso
```

3.13.2 Redfish* Based

```
SDPTool <ipv4> <username> <password> vmedia <smb://user:pass@host/file_name.ISO> [-no_user_interaction]
```

```
SDPTool <ipv4> <username> <password> vmedia <nfs://host/file_name.ISO>[-no_user_interaction]
```

The vmedia command allows the addition of virtual media in .iso format. The vmedia command supports SMB and NFS protocols. The shares need to be created and configured beforehand. The SDP Tool does not create or configure them. Add the relevant virtual media by redirecting the ISO file specified. Only one .iso can be mounted at a time.

Example: SDPTool 192.168.1.10 admin admin123 vmedia smb://user:pass@host/share_dir/image.iso

Example: SDPTool 192.168.1.10 admin admin123 vmedia nfs://host/share_dir/image.iso

Note: This Redfish*-based operation is only supported on the following Intel product families:

- Intel® Server S2600WF and S2600WFR Families
 - Intel® Server S2600ST and S2600STR Families
 - Intel® Server S2600BP and S2600BPR Families
 - Intel® Server S9200WK Family
 - Intel® Server D50TNP
 - Intel® Server M50CYP
 - Intel® Server D40AMP
 - Intel® Server M50FCP
 - Intel® Server D50DNP
-

3.13.3 Config File Based

```
SDPTool <ipv4> <username> <password> vmedia -c <config file> [-no_user_interaction]
```

This option provides the user the ability to provide a configuration file instead of image file. The creation of the configuration file is covered in [Section 2.5](#). The configuration file is used to pick the predefined image file for virtual media mount.

3.14 IPMI

SDPTool <ipv4> <username> <password> ipmi <ipmitool arguments>

The IPMI command is followed by arguments allowing the execution of ipmitool supported commands.

Example: SDPTool 192.168.1.10 admin admin123 ipmi lan print 3

3.15 Power

SDPTool <ipv4> <username> <password> power <status | on | off | cycle | reset>

-force

The power command returns, and can set, the power status of a server.

Example: SDPTool 192.168.1.10 admin admin123 power status

Use -force when you want to force the power operation to occur, which is generally not allowed when an update is being performed. A typical use case is to reboot the system when staging a cup deploy in attached mode.

3.16 Sensor

SDPTool <ipv4> <username> <password> sensor

The sensor command displays the relevant sensor information of a server.

Example: SDPTool 192.168.1.10 admin admin123 sensor

3.17 SEL

SDPTool <ipv4> <username> <password> sel [-f <filename to save sel-log>] [-c] [-w] [-i]

This command retrieves the SEL log.

Options:

-f = specifies a file name to save the SEL log.

-i = information

-c = critical

-w = warning

Example: SDPTool 192.168.1.10 admin admin123 sel -w -l -f save_log.txt

3.18 Set LAN

Deprecated: *This is a Deprecated command, and no further support will be provided*

SDPTool <ipv4/ipv6> <username> <password> setlan <channel> <ipv4> <mask> <gateway> <primary dns> <secondary dns>

The setlan command configures the BMC LAN IP IPV4 address of a particular LAN channel.

Example: SDPTool 192.168.1.10 admin admin123 setlan 2 192.168.1.12 255.255.255.0 192.168.1.1 8.8.8.8 0.0.0.0

Note: Not supported from Eagle Stream generation onwards.

3.19 Disable LAN

Deprecated: *This is a Deprecated command, and no further support will be provided*

SDPTool <ipv4/ipv6> <username> <password> disablelan <channel>

The disablelan command disables a BMC LAN IP IPV4 address of a particular LAN channel.

Example: SDPTool 192.168.1.10 admin admin123 disablelan 2

Note: Not supported from Eagle Stream generation onwards.

3.20 Set LAN IPV6

Deprecated: *This is a Deprecated command, and no further support will be provided*

For S2600WT/S2600WTR/S2600KP/S2600KPR/S2600TP/S2600TPR/S2600CW/S2600CWR:

```
SDPTool <ipv4/ipv6> <username> <password> setlanipv6 <channel> <ipv6> <prefix  
length[32|64|128]> <ipv6 gateway>
```

The setlanipv6 command configures the BMC LAN IP IPV6 address of a particular LAN channel.

Example: SDPTool 192.168.1.10 admin admin123 setlanipv6 2 fe80::12 64 fe80::1

For S2600WF/S2600WFR/S2600SW/S2600SWR/S2600BP/S2600BPR/D50TNP/M50CYP/D40AMP:

```
SDPTool <ipv4/ipv6> <username> <password> setlanipv6 <channel> <ipv6> <prefix  
length[32|64|128]> <ipv4/6 gateway> <ipv4/6 primary dns> <ipv4/6 secondary dns>
```

The setlanipv6 command configures the BMC LAN IP IPV6 address of a particular LAN channel.

Example: SDPTool 192.168.1.10 admin admin123 setlanipv6 2 fe80::12 64 192.168.1.1 0.0.0.0 0.0.0.0

For M70KLP/M20NTP:

```
SDPTool <ipv4/ipv6> <username> <password> setlanipv6 <channel> <ipv6> <prefix  
length[32|64|128]> <ipv6 gateway> <ipv6 primary dns> <ipv6 secondary dns>
```

The setlanipv6 command configures the BMC LAN IP IPV6 address of a particular LAN channel.

Example: SDPTool 192.168.1.10 admin admin123 setlanipv6 2 fe80::12 64 fe80::1 fe80::0 fe80::0

Note: Not supported from Eagle Stream generation onwards.

3.21 Disable LAN IPV6

Deprecated: *This is a Deprecated command, and no further support will be provided*

```
SDPTool <ipv4/ipv6> <username> <password> disablelanipv6 <channel>
```

The disablelanipv6 command disables the BMC IPV6 LAN of a particular LAN channel.

Example: SDPTool 192.168.1.10 admin admin123 disablelanipv6 2

Note: Not supported from Eagle Stream generation onwards.

3.22 LAN Fail Over

Deprecated: *This is a Deprecated command, and no further support will be provided*

```
SDPTool <ipv4> <username> <password> failover < status | enable | disable>
```

The failover command returns, sets, and disables LAN failover.

Example: SDPTool 192.168.1.10 admin admin123 failover status

Note: Not supported from Eagle Stream generation onwards.

3.23 Node Position

Deprecated: *This is a Deprecated command, and no further support will be provided*

SDPTool <ipv4> <username> <password> nodeposition

The nodeposition command displays node position within a chassis, and only supports a half-width SKU.

Note: Support is available for select multi-node systems. Not supported from Eagle Stream generation onwards.

Example: SDPTool 192.168.1.10 admin admin123 nodeposition

3.24 System Information

SDPTool <ipv4> <username> <password> systeminfo

The systeminfo command displays the system information related to the BMC and baseboard including the BMC version, BIOS version, RMM, and others.

Example: SDPTool 192.168.1.10 admin admin123 systeminfo

3.25 FRU

SDPTool <ipv4> <username> <password> fru {print | set <param> <value>}

The fru command displays or sets available FRU information.

Example: SDPTool 192.168.1.10 admin admin123 fru print

To set FRU:

Example: SDPTool 192.168.1.10 admin admin123 fru set <param> <value>

3.26 Memory Information

SDPTool <ipv4> <username> <password> memoryinfo

The memoryinfo command displays relevant memory information.

Example: SDPTool 192.168.1.10 admin admin123 memoryinfo

3.27 CPU Information

SDPTool <ipv4> <username> <password> cpuinfo

The cpuinfo command displays relevant CPU information.

Example: SDPTool 192.168.1.10 admin admin123 cpuinfo

3.28 Memory Temperature

SDPTool <ipv4> <username> <password> memorytemp

The memorytemp command displays the temperature of the system memory.

Example: SDPTool 192.168.1.10 admin admin123 memorytemp

3.29 Power Statistics

SDPTool <ipv4> <username> <password> powerstatistics

The powerstatistic command displays system power statistics.

Example: SDPTool 192.168.1.10 admin admin123 powerstatistics

3.30 Set LAN DHCP

Deprecated: *This is a Deprecated command, and no further support will be provided*

SDPTool <ipv4/ipv6> <username> <password> setlandhcp <channel>

The setlandhcp command sets the BMC LAN IPV4 to the DHCP of a particular LAN channel.

Example: SDPTool 192.168.1.10 admin admin123 setlandhcp 2

Note: Not supported from Eagle Stream generation onwards.

3.31 Set LAN DHCP IPV6

Deprecated: *This is a Deprecated command, and no further support will be provided*

SDPTool <ipv4/ipv6> <username> <password> setlandhcpipv6 <channel>

The setlandhcpipv6 command sets the BMC LAN IPV6 to the DHCP of a particular LAN channel.

Example: SDPTool 192.168.1.10 admin admin123 setlandhcpipv6 2

Note: Not supported from Eagle Stream generation onwards.

3.32 Set LAN Stateless ICMPV6

Deprecated: *This is a Deprecated command, and no further support will be provided*

SDPTool <ipv4/ipv6> <username> <password> setlanicmpv6 <channel>

The setlanicmpv6 command sets the BMC LAN IPV6 to the stateless ICMP.

Note: This operation is only supported on the following Intel product families:

- Intel® Server S2600WT and S2600WTR Families
 - Intel® Server S2600KP and S2600KPR Families
 - Intel® Server S2600TP and S2600TPR Families
 - Intel® Server S2600CW and S2600CWR Families
 - Intel® Server S9200WK Family
-

Example: SDPTool 192.168.1.10 admin admin123 setlanicmpv6 2

Note: Not supported from Eagle Stream generation onwards.

3.33 Debug Log

SDPTool <ipv4> <username> <password> debuglog <filename> [-force]

The debuglog command retrieves the BMC debug log file in .zip file format.

Example: SDPTool 192.168.1.10 admin admin123 debuglog debug_log.zip

- [-force]: Forces the BMC transfer mode to exit when the command is executed.

3.34 Supported Updates

SDPTool <ipv4> <username> <password> supportedupdates -c <config_file>

The supportedupdates command lists the type of updates available with the configuration file for the Managed Server. The result is one of the following values: None, SUP ONLY, Custom ONLY, Both.

3.35 Unmount

3.35.1 IPMI Based

SDPTool <ipv4> <username> <password> unmount [<IMAGE/ISO> | -clean] [-no_user_interaction]

Note: The -clean removes everything, including the active connections. Users are advised caution.

The unmount command allows the user to unmount or remove media that was mounted to the Managed Server using the vmedia command. The image/iso is the full path to the image that was mounted.

Note: This operation is only supported on the following Intel product families:

- Intel® Server S2600WF and S2600WFR Families
 - Intel® Server S2600ST and S2600STR Families
 - Intel® Server S2600BP and S2600BPR Families
 - Intel® Server S9200WK Family
 - Intel® Server M70KLP Family
 - Intel® Server D50TNP
 - Intel® Server M50CYP
 - Intel® Server D40AMP
 - Intel® Server M20NTP
-

3.35.2 Redfish* Based

SDPTool <ipv4> <username> <password> unmount

The unmount command allows the unmounting of virtual media mounted previously in .iso format only from the remote machine samba share via Redfish*.

Example: SDPTool 192.168.1.10 admin admin123 unmount

3.36 Storage Information

SDPTool <ipv4> <username> <password> storageinfo

The storageinfo command displays information about the storage devices present on the Managed Server that SDP Tool can communicate with, providing details regarding firmware version, device name, and others.

Example: SDPTool 192.168.1.10 admin admin123 storageinfo

3.37 NIC Information

SDPTool <ipv4> <username> <password> nicinfo

The nicinfo command displays information about the network(s) present on the Managed Server that SDP Tool can communicate with, providing details regarding firmware version, MAC address, and others.

Example: SDPTool 192.168.1.10 admin admin123 nicinfo

3.38 Online Firmware (BIOS/BMC/ME/FRUSDR/DCPMM/SSD) Version Check

SDPTool <ipv4> <username> <password> softwareinventory

This command shows both online and backup version on the system.

Examples: SDPTool 192.168.1.10 admin admin softwareinventory

SDPTool softwareinventory <Managed Serverid> or

SDPTool softwareinventory <SSD Model>

These commands only show online versions for the given Managed Server or SSD model.

Examples:

SDPTool softwareinventory S2600WFQ or

SDPTool softwareinventory SSDPE2KX040T8

3.39 Custom Update Package Inspect

SDPTool <ipv4> <username> <password> cup_inspect [<cup-file.zip>] [--format <filename.json>]

The cup_inspect command is used to check the status of a staged cup_deploy and performs a compliance check against a given CUP. The command will provide details to the user including the current installed versions of the firmware and the version available in the cup package.

Sample output: 1

Component	Installed Version	CUP Version	State
BIOS	01.01.0005	01.01.0005	Compliant
SDR	0.40	0.40	Compliant
ME	04.04.04.62	04.04.04.62	Compliant
BMC	2.88.e5f45b9c	2.88.e5f45b9c	Compliant
CPLD	3.5	3.5	Compliant
PMEM	-	2.2.0.1553	Staged (reboot required)
NIC: X550T2	-	3.60	Staged (reboot required)
RAID: RSP3MD088F	-	51.20.0-4342	Staged (reboot required)
SSD: SSDSCKKB240G801	-	XC311140	Staged (reboot required)

Sample output: 2

Component	Installed Version	CUP Version	State
BIOS	01.01.0005	01.01.0005	Compliant
SDR	0.40	0.40	Compliant
ME	04.04.04.62	04.04.04.62	Compliant
BMC	2.88.e5f45b9c	2.88.e5f45b9c	Compliant
CPLD	3.5	3.5	Compliant
PMEM	-	2.2.0.1553	Update in progress
NIC: X550T2	-	3.60	Update in progress
RAID: RSP3MD088F	-	51.20.0-4342	Update in progress
SSD: SSDSCKKB240G801	-	XC311140	Update in progress

PTO

Sample output: 3

Component	Installed Version	CUP Version	State
BIOS	01.01.0005	01.01.0005	Compliant
SDR	0.40	0.40	Compliant
ME	04.04.04.62	04.04.04.62	Compliant
BMC	2.88.e5f45b9c	2.88.e5f45b9c	Compliant
CPLD	3.5	3.5	Compliant
PMEM	-	2.2.0.1553	-
NIC: X550T2	3.60	3.60	Compliant
RAID: RSP3MD088F	51.20.0-4342	51.20.0-4342	Compliant
SSD: SSDSCKKB240G801	-	XC311140	-

The output of cup_inspect may indicate one of many states; the following list contains the states and what they mean:

- **Staging:** CUP is being uploaded, will remain in this state till the upload is completed.
- **Staged (reboot required):** CUP is uploaded and will be applied in the next reboot (waiting on user to reboot the Managed Server)
- **Update in progress:** CUP has been deployed and the update is in progress.
- **Compliant:** The component is now compliant or same as the one in the CUP provided.
- **Non-Compliant:** The component is not compliant with the CUP provided.

- -: The component is not available or not determined.

cup_inspect will also fetch the logs that are available on the system in the case of cup_deploy in detached mode, more about detached mode in Section 3.40.

--format <filename.json> will generate a file (filename.json in json format) that will have the formatted/programmable output of the cup_inspect command.

Note: Current implementation requires the ipmi port to be exempted in the firewall rules.

3.39.1 Config File Based

SDPTool <ipv4> <username> <password> cup_inspect -c <config file>

This option provides the user the ability to provide a configuration file instead of the CUP folder path. The creation of the configuration file is covered in [Section 2.5](#). The configuration file is used to provide the necessary information, such as CUP folder, format json for the Managed Server being updated.

3.40 Custom Update Package Deploy

The custom update package (CUP) deployment feature performs a customized update, which is defined by the user. The CUP can be configured and downloaded from the Intel portal. Refer to [Chapter 6](#) for more details. Additional logs are generated for cup_deploy and can be found in the location:

/usr/local/log/SDPTool/Logfiles/IP_Address/cup_deploy/

Usually, CUP is configured to bundle the following firmware:

1. Onboard components BIOS/BMC/SDR/ME/CPLD
2. Firmware for peripherals such as NIC, HBA, and SSD

SDPTool <ipv4> <username> <password> cup_deploy <CUP zip> [-staged] [-attached/-detached][-no_user_interaction]

Example: SDPTool 192.168.1.10 admin admin123 cup_deploy /root/cup.zip -no_user_interaction

-staged option is to be used if the user intends to perform staging or deferred update of the cup package. During the staging process SDP Tool moves the cup to the Managed Server and waits for the user to reboot the system, upon reboot the staged cup package will automatically update all the staged components.

Note: The staging location and its contents are wiped if the BMC is reset.

3.40.1 CUP Deploy modes

There are two modes of update in the -staged solution:

3.40.1.1 Attached mode

In attached mode, SDP Tool will wait for the Managed Server to be reset. Resetting or Rebooting is the responsibility of the User.

3.40.1.2 Detached mode

Detached mode is a silent update. It is a non-blocking mode where SDP Tool returns to the prompt after the staging activity is completed. The update occurs in the next reboot of the system. Since this is a non-blocking call no logs are accumulated other than the output on the shell. Please refer to Section 3.38 which provides the details of cup_inspect which is used to retrieve the logs in the detached scenario.

Note: Some intermittent failures are seen using experimental packages (related to SSD) in CUP, a retry should fix the issue. Current implementation requires the IPMI port to be exempted in the firewall rules. IPMI ports are used to update SEL logs with the progress.

3.40.2 Config File Based

SDPTool <ipv4> <username> <password> cup_deploy -c <config file> [-no_user_interaction]

This option provides the user the ability to provide a configuration file instead of the CUP folder path. The creation of the configuration file is covered in [Section 2.5](#). The configuration file is used to provide the necessary information, such as CUP folder, mode for the Managed Server being updated.

Chapter 4. custom_deploy

The custom_deploy allows the users to customize or script the actions they want to perform on the Managed Server. The script runs on the EFI Shell, ensuring that the user can write scripts to perform actions based on individual requirements.

When creating a custom package, the user must write a .nsh file and name it deploy.nsh. When executing custom_deploy as described in [Section 3.3.5](#), the SDP Tool looks for the deploy.nsh file within a folder that the user provides as a command-line argument.

The custom folder provided by the user is used to create an image with standard EFI applications and remotely mount it onto the Managed Server. The Managed Server then executes the deploy.nsh script to perform the necessary actions and return the results/outputs to the user.

While writing the deploy.nsh file, the user can also redirect the output to two distinct files from which the SDP Tool collects logs:

- deploy_result.log
- deploy_details.log

The deploy_result.log can be used to redirect the success or failure status of a particular action. If this redirection is used properly, all successes/failures will be captured in the custom_deploy_result.txt log file.

The deploy_details.log can have the detailed log of every operation in the deploy.nsh and is available as custom_deploy_details.txt.

Notes:

- The output of the deploy.nsh file is also captured in a separate file named custom_deploy_output.txt that is available with the other logs in the standard location that is described in [Section 3.1](#).
-

In the following deploy.nsh file, actions are logged. The deploy.nsh file can be used to perform complicated operations as well and is dependent on the user's requirement.

```
echo "Starting the user custom operation" >> deploy_details.log
ls
echo "listing files : successful" >> deploy_result.log
```

The user can write deploy.nsh scripts to perform firmware updates of various components, which include, but are not limited to SDD, HBA, and NIC.

Notes:

- Since the custom folder is provided as an argument, it is the only folder included in the image mounted to the Managed Server. If any additional binary is needed when calling from the deploy.nsh file, it must be available within the custom folder before the custom deploy is invoked.
-

Chapter 5. SDP Tool Configuration file

A configuration file can be used to set some of the operational values for the SDP Tool. The configuration files can be found in the location `/etc/sdptool/sdptool.conf`. The current supported values and a sample Configuration file is as follows:

```
[Global]
; SDPTool version
version=4.2-0

; Establish secure connections only
verifyCertificate=false

; SDPTool log directory
logRootPath=/usr/local/log/SDPTool/Logfiles

; Enable full log
enableFullLog=true

[cup_deploy]
staging=false
; Staging directory for CUP deploy
stagePath=/var/sdptool/cup
```

- `verifyCertificate`
 - This variable is used to establish a secure connection with BMC by verifying the BMC's certificate.
 - User will have to upload a certificate to the BMC EWS (Embedded Web Server) for this to be effective.
 - If no certificate is available with BMC EWS, SDP Tool will error out.
- `logRootPath`
 - This variable is used set the base path where log files will be created.
- `enableFullLog`
 - A historic log with chronological events will be logged.
 - Primarily used for debugging and auditing.
- `staging`
 - Allows SDP Tool to stage a CUP on the target Managed Server.
 - CUP based updates will be performed upon the next reboot.

Note: Variables must be provided in their respective sections.

Chapter 6. CUP

The cup_deploy requires a specific package called as CUP (custom update package). The CUP can be created and downloaded using the following steps (additional information is available at <https://servertools.intel.com/tools/firmware-tracker/>):

1. Register with the server tools program.
2. After the registration is complete, go to: <https://servertools.intel.com/firmware-tracker/>.
3. Search for the product codes for the desired product. L9 product codes or individual product codes are accepted. Select **CUP Download** after the search query is complete. Refer to Figure 3.
4. Choose the required packages from the list.
5. Once one or more packages are selected, the **Submit** button becomes active. Click it to download the Custom Update Package. Refer to Figure 4.

The following figures show screenshots of the CUP download in action.

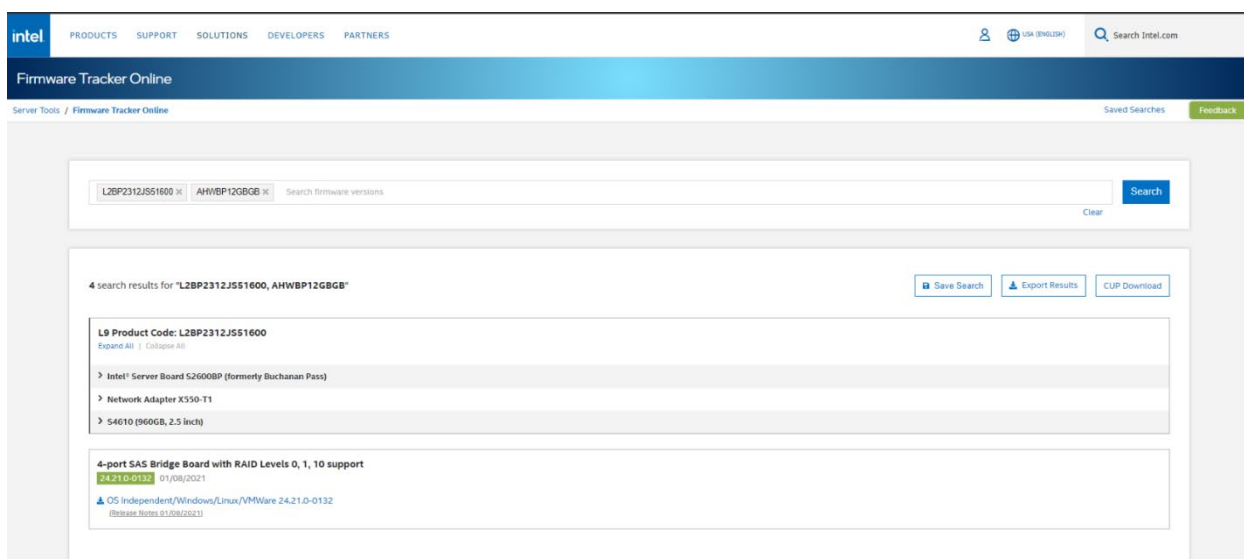


Figure 3. Server Tool Search Results

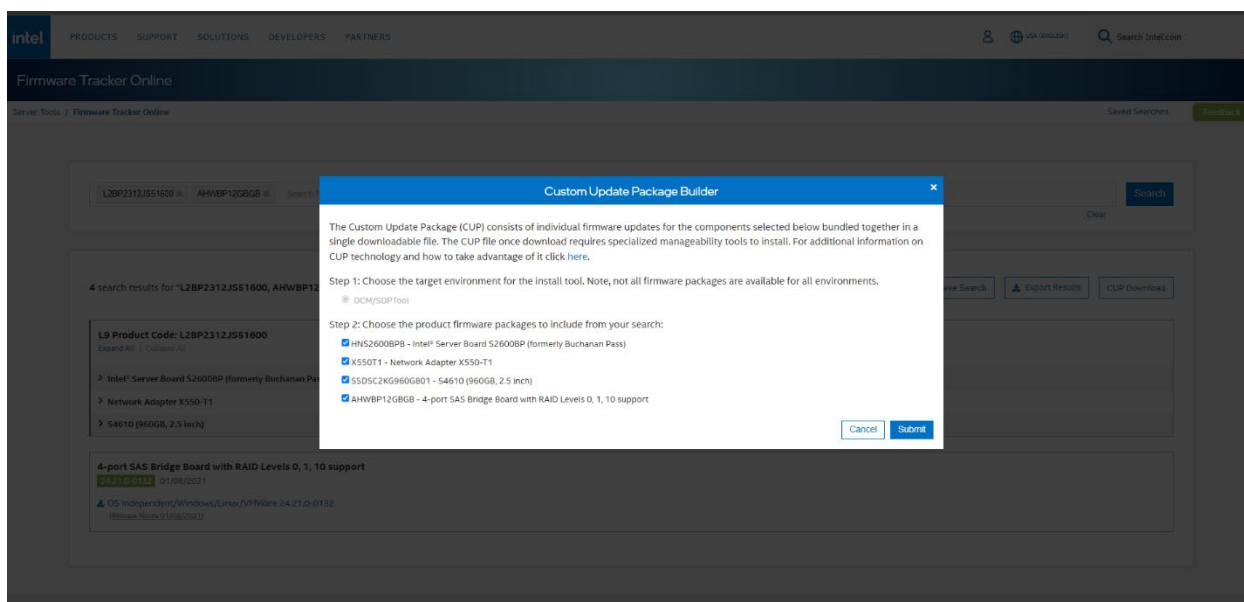
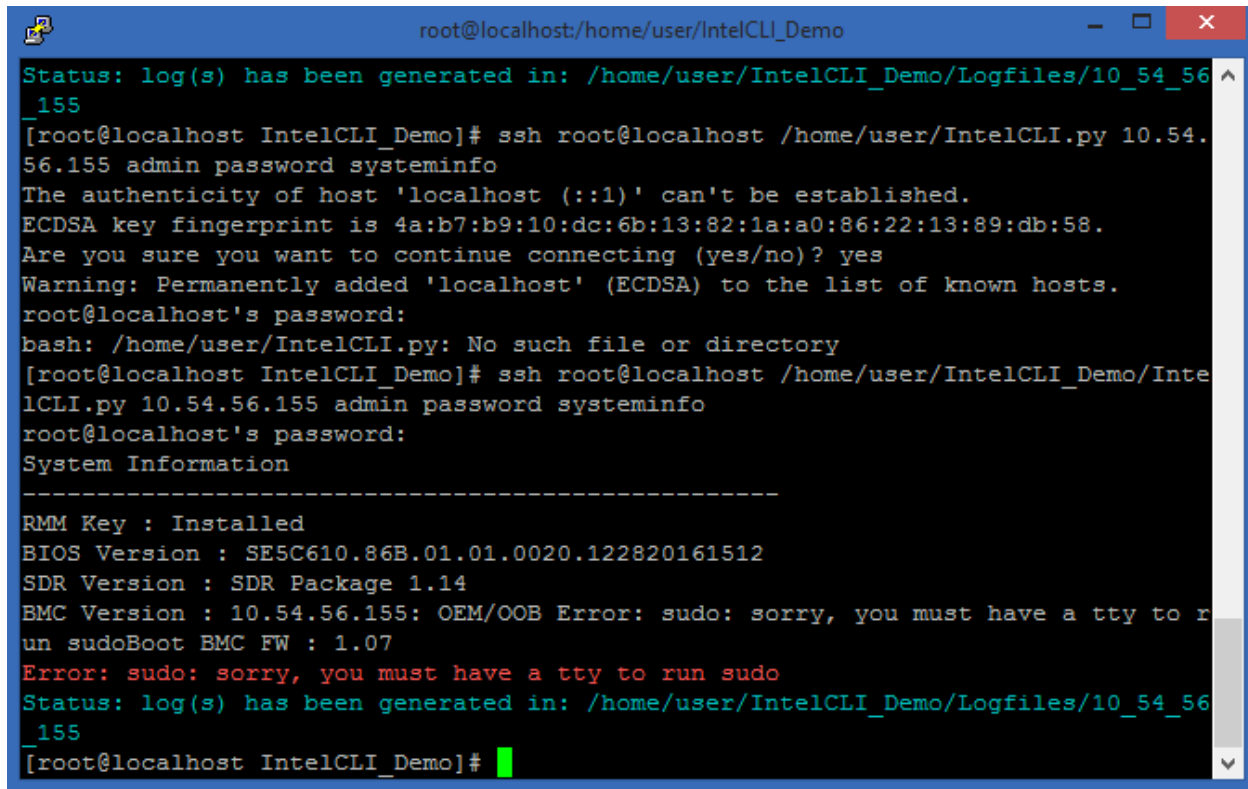


Figure 4. CUP Download

Chapter 7. Troubleshooting Tips

This section lists the known possible errors encountered during the use of this product with workarounds to address these errors.

7.1 SSH Command "sudo: you must have a tty to run sudo" Error



```
root@localhost:/home/user/IntelCLI_Demo
Status: log(s) has been generated in: /home/user/IntelCLI_Demo/Logfiles/10_54_56_155
[root@localhost IntelCLI_Demo]# ssh root@localhost /home/user/IntelCLI.py 10.54.56.155 admin password systeminfo
The authenticity of host 'localhost (:::1)' can't be established.
ECDSA key fingerprint is 4a:b7:b9:10:dc:6b:13:82:1a:a0:86:22:13:89:db:58.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'localhost' (ECDSA) to the list of known hosts.
root@localhost's password:
bash: /home/user/IntelCLI.py: No such file or directory
[root@localhost IntelCLI_Demo]# ssh root@localhost /home/user/IntelCLI_Demo/IntelCLI.py 10.54.56.155 admin password systeminfo
root@localhost's password:
System Information
-----
RMM Key : Installed
BIOS Version : SE5C610.86B.01.01.0020.122820161512
SDR Version : SDR Package 1.14
BMC Version : 10.54.56.155: OEM/OOB Error: sudo: sorry, you must have a tty to run sudoBoot BMC FW : 1.07
Error: sudo: sorry, you must have a tty to run sudo
Status: log(s) has been generated in: /home/user/IntelCLI_Demo/Logfiles/10_54_56_155
[root@localhost IntelCLI_Demo]#
```

Figure 5. SSH Command Error

To direct using the ssh command, add arg `-t`.

Example: `ssh -t root@localhost SDPTool 192.168.1.10 admin admin123 powerstatistics`

7.2 Tar Time Stamp Messages

```
tar: cmdtoolX64.efi: time stamp 2017-04-12 22:37:02 is 115999883.204964241 s in the future
tar: efifmt.efi: time stamp 2017-04-12 22:37:22 is 115999903.20435322 s in the future
tar: ipmi.efi: time stamp 2017-04-13 17:23:02 is 116067443.203744765 s in the future
tar: mkedk2ramdiskX64.efi: time stamp 2017-04-12 22:37:42 is 115999923.20330491 s in the future
tar: mkramdiskX64.efi: time stamp 2015-02-05 15:55:54 is 47115015.202896016 s in the future
tar: RamDiskDxe.efi: time stamp 2017-04-14 02:56:44 is 116101865.202739364 s in the future
tar: ramdisk.efi: time stamp 2015-02-05 15:55:54 is 47115015.202524166 s in the future
tar: rundeploy.nsh: time stamp 2017-04-18 17:27:00 is 116499681.202410696 s in the future
tar: Startup.nsh: time stamp 2017-04-18 17:22:02 is 116499383.202309794 s in the future
tar: syscfg.efi: time stamp 2017-02-27 21:05:30 is 112192791.201309007 s in the future
tar: syscfg_temp.efi: time stamp 2017-02-27 21:05:32 is 112192793.18213325 s in the future
tar: vmdrive_map: time stamp 2017-03-16 21:53:50 is 113664491.18204245 s in the future
```

Figure 6. Tar Time Stamp Messages

These messages do not harm the system. To avoid seeing these messages, ensure that the date and time of the managed system are correct.

7.3 Java* Version Required (Java* 1.7)



Figure 7. Java* Version

OpenJDK* or Oracle Java* (version 1.7 onwards) is required to run:

- kvm
- update
- customdeploy
- setoptions
- deployoptions
- getbiosoptions
- getini

To check the Java* version and provider, run:

```
Prompt #> java -version
```

7.4 Proxy Settings

Proxy settings may be required to connect to the external pip repository during installation time. The defusedxml module is required for using the SDP Tool functionality that uses configuration files (supported updates and update). Update only while using -c option; else, the defusedxml module is not required.

```
Prompt #> export http_proxy="<proxy address:proxy port>"
```

```
Prompt #> export https_proxy="<proxy address:proxy port>"
```

Set both the environment variables appropriately for the Management Server's network environment.

Chapter 8. Error Codes

Table 4. Error Codes

Error Code	Error Type	Error Description
0	NoError	Success / no failure
1	ENoRMM	RMM module absent
2	ENoIPMI	IPMI module absent
3	ENoFileCreate	Error creating a file
4	ENoRetrieve	Error in retrieving the data
5	ENoProdRetrieve	Error retrieving the product information
6	ENoSoftReset	Error trying to soft-reset
7	ENoJava	Error Java* not present
8	ENoPerm	Error not permitted/ need privileges
9	ENoConnect	Error trying to connect to the system
10	ENoRedirection	Error redirecting the image
11	ENotSupported	Error not supported currently
12	EUnsupportedPlatform	Platform not supported
13	EUnsupportedOperation	Operation not supported
14	ECurrNotSupported	Error currently not supported
15	EMissingFiles	Missing required files
16	EMissingTags	Missing required tag in XML
17	EMissingHardware	Missing required hardware
18	EMissingArgs	Missing required arguments
19	EMissingTools	Missing required tools
20	EInterrupt	Process interrupted
21	EInvalidArgs	Invalid arguments
22	EInvalidIP	Invalid IP
23	EInvalidChannel	Invalid channel
24	EInvalidSubnet	Invalid Subnet mask
25	EInvalidFilename	Invalid file name
26	EInvalidFileExt	Invalid/unexpected file extension
27	EInvalidPath	Invalid path
28	EInvalidSMBIOS	Invalid/unsupported BIOS region
29	EMismatchIPGW	IP address and gateway are a mismatch
30	EIPMICmdError	Error running IPMI command
31	EIPMICmdTimeout	Error IPMI command timed out
32	EDupVMCLI	VMCLI already running, duplicate error
33	EMount	Error mounting / unmounting the image
34	EDataConvr	Error converting data

Error Code	Error Type	Error Description
35	EKVMSessFull	Error launching KVM session is full
36	EUnknown	Unknown error
37	ESetoptionSupport	Error trying to set option
38	EOperationFail	Operation fails or reports error
39	EcurlCmd	Error from curl process
40	ESubprocess	Error invoking the process
41	ECleanupImage	Error cleaning up an Image
42	ETermDefunc	Error terminating a defunct process
43	ETermSuspend	Error terminating a suspended process
44	EKillCmd	Error trying to kill a command
45	EStartVMCLI	Unable to start VMCLI
46	ETestapp	Testapp hits error
47	ESUPTooLarge	SUP package provided is too large
48	ESetTransMode	Error setting transfer mode in BMC
49	ESingleFile	Error in single file
50	EFileNotFound	File / path not found
51	ESystemError	System gives error
52	EAbort	Aborted
53	ESysCfg	Error in using the Intel® Server Configuration Utility
54	ERedfish	Redfish* general error
55	ETimedOut	Timed out in trying to establish connection
56	ERedfishSession	Failure in creating Redfish* session
57	ERedfishResponse	Redfish* HTTP response error
58	ERedfishNoSensor	Sensor not present
59	ERedfishAuthorization	Wrong username or password or hostname given
60	ERedfishUnexpected	Unexpected HTTP Redfish* error
61	ERedfishRedirect	Redirection error
62	ERedfishAttribute	JSON key is not present
63	EInvalidLength	Error in FRU fields length
64	ERedfishSchema	Error in Redfish* schema
65	EVMRedirect	Error in virtual media redirection
66	ERecursionLimit	Limit the number of recursive calls to five at most
67	ENoStorage	No storage device present
68	EFwTrack	Fwtrack is not installed or available
69	EScraper	No fwtrack data available
70	ERedfishBMCWaitAbort	Failed to reach BMC via Redfish* API
71	EInvalidConfigFile	JSON configuration file is invalid
72	EConfigFileRead	JSON configuration file read error

Error Code	Error Type	Error Description
73	EMissingSKUConfigFile	Missing SKU information in configuration file
74	EMissingPackageConfigFile	SKU package information is not present in configuration file
75	EMissingUnmountImageConfigFile	Unmount image path information is not present in the configuration file

Table 5. IPMI Reboot Features List

Command	Reboot Required
cpuinfo	No
custom_deploy	Yes
debuglog	No
deployoptions	Yes
disablelan	No
disablelanipv6	No
failover	No
fru	No
getbiosoptions	Yes
getini	Yes
ipmi	N/A
kvm	No
memoryinfo	No
memorytemp	No
nodeposition	No
power	No
powerstatistics	No
sel	No
sensor	No
setlan	No
setlandhcp	No
setlandhcpipv6	No
setlanicmpv6	No
setlanipv6	No
setoptions	Yes
supportedupdates	No
systeminfo	No
unmount	No
update	Yes
vmedia	No
softwareinventory	No

Chapter 10. Redfish* Features List

Table 6. Redfish* Features List

Command	Support	Reboot Required
cpuinfo	Yes	No
debuglog	Partial ¹	No
set_biosconfig_all	Yes	Yes
disablelan	No	No
disablelanipv6	No	No
failover	No	No
fru	Partial ²	No
get_biosconfig	Yes	No
get_biosconfig_all	Yes	No
ipmi	No	N/A
kvm	No	No
memoryinfo	Yes	No
memorytemp	Yes	No
nodeposition	No	No
power	Yes	No
powerstatistics	Yes	No
sel	Yes	No
sensor	Yes	No
setlan	No	No
setlandhcp	No	No
setlandhcpipv6	No	No
setlanicmpv6	No	No
setlanipv6	No	No
set_biosconfig	Yes	Yes
supportedupdates	No	No
systeminfo	Yes	No
unmount	Yes	No
update	Yes	Yes
vmedia	Yes	No
storageinfo	Yes	No
nicinfo	Yes	No
softwareinventory	No	No
cup_deploy	Yes	Yes

Notes: (1) Only the mechanism to generate the debuglog file at the target machine is supported by Redfish* currently. There is no way to check if the target machine is in transfer mode from beforehand or to fetch the generated debuglog file once it is generated to the local machine. It is still happening via IPMI. (2) Redfish* currently only provides option to fetch the FRU components' information and hence fru print is supported by Redfish* but fru set is not and still needs IPMI.

Chapter 11. Appendix A. Unsupported command list

Table 7. Unsupported command list

Command	Unsupported whole command	Unsupported arguments	Comments
setlan	Yes		It is unsupported from Eagle Stream generation onwards.
disablelan	Yes		It is unsupported from Eagle Stream generation onwards.
setlanipv6	Yes		It is unsupported from Eagle Stream generation onwards.
disablelanipv6	Yes		It is unsupported from Eagle Stream generation onwards.
filover	Yes		It is unsupported from Eagle Stream generation onwards.
nodeposition	Yes		It is unsupported from Eagle Stream generation onwards.
setlandhcp	Yes		It is unsupported from Eagle Stream generation onwards.
setlandhcpipv6	Yes		It is unsupported from Eagle Stream generation onwards.
setlanicmpv6	Yes		It is unsupported from Eagle Stream generation onwards.
kvm	Yes		It is unsupported from Eagle Stream generation onwards.
update	No	softreset	It is unsupported from Eagle Stream generation onwards.
getbiosoptions	No	softreset	It is unsupported from Eagle Stream generation onwards.
softwareinventory	Yes		The server will be closed on 2024/06/01 and services will no longer be provided.

Chapter 12. Appendix B. Unsupported command list in IPMI based

Table 8. Unsupported command in IPMI based

Command	Supported in IPMI based	Comments
vmedia	No	It is unsupported from Eagle Stream generation onwards.

Chapter 13. Appendix C. Removed package

1. List of removed tools

- VMCLI

Note: VMCLI tool is removed from SDPTool version 4.2-2
