Dell EMC PowerSwitch S5200-ON Series BMC User Guide

March 2021

March 2021
Rev. A03

Copyright

© 2018 - 2021 Dell Inc. or its subsidiaries. All rights reserved. Dell, EMC, and other trademarks are trademarks of Dell Inc. or its subsidiaries. Other trademarks may be trademarks of their respective owners.

Contents

Chapter 1: About this guide **4**
Information symbols 4
Document revision history 5

Chapter 2: Hardware and software support **6**

Chapter 3: Configuration methods **7**
Configurations 9
Date and time 10
SNMP and email alerts 11
Add and delete users 13
Firewall 17
Event log 28
Default configuration restore 29

Chapter 4: Host power control **31**

Chapter 5: Firmware update **32**

Chapter 6: Access system health sensors **33**

Chapter 7: IPMI commands **35**

Chapter 8: ipmiutil package **38**

Chapter 9: Access FRU data **39**

Chapter 10: Dell EMC support **41**

1

About this guide

This guide provides information for using the Dell EMC baseboard management controller (BMC).

CAUTION: To avoid electrostatic discharge (ESD) damage, wear grounding wrist straps when handling this equipment.

NOTE: Only trained and qualified personnel can install this equipment. Read this guide before you install and power on this equipment. This equipment contains two power cables. Disconnect both power cables before servicing.

NOTE: This equipment contains optical transceivers, which comply with the limits of Class 1 laser radiation.

Figure 1. Class 1 laser product tag

NOTE: When no cable is connected, visible and invisible laser radiation may be emitted from the aperture of the optical transceiver ports. Avoid exposure to laser radiation. Do not stare into open apertures.

Language

NOTE: This guide may contain language that is not consistent with the current guidelines. Dell EMC plans to update the guide over subsequent releases to revise the language accordingly.

Topics:

- Information symbols
- Document revision history

Information symbols

This book uses the following information symbols:

NOTE: The **Note** icon signals important operational information.

CAUTION: The Caution icon signals information about situations that could result in equipment damage or loss of data.

WARNING: The Warning icon signals information about hardware handling that could result in injury.

WARNING: The ESD Warning icon requires that you take electrostatic precautions when handling the device.

Document revision history

Table 1. Revision history

Revision	Date	Description
A00	2018-09	Initial release
A01	2019-03	Updated to include the S5224F-ON and S5212F-ON platforms.
A02	2021-02	Removed broken link. Updated document to current IDD standards. Added language note. Updated the *Default configuration restore* section. Added the *ipmi commands* section.
A03	2021-03	Fixed incorrect link.

2

Hardware and software support

For the most current BMC update information, see the *S5200-ON Series Release Notes*.

For more information about the intelligent platform management interface (IPMI), see the IPMI resources that is hosted by Intel at https://www.intel.com/content/www/us/en/servers/ipmi/ipmi-technical-resources.html.

NOTE: The BMC out-of-band (OOB) network or LAN is not enabled for Trade Agreement Act-qualified (TAA) switches. The BMC OOB is enabled for non-TAA-qualified switches.

Required drivers

In Linux, the baseboard management controller (BMC) uses the `ipmitool` open-source tool during testing. To configure or get data from the BMC, `ipmitool` sends `ipmi` commands to the BMC. You must have the IPMI driver installed to use `ipmitool`.

To access `ipmitools`, go to https://sourceforge.net, search for `ipmitools`, then select the **See Project** button.

NOTE: Although there are newer versions available, the `ipmitool` and driver versions used during testing the BMC are:

- Linux version: 4.9.30
- `ipmitool` version: 1.8.18
- `ipmi` driver that the `ipmitool` uses is built with kernel 4.9.30.

BMC access

Access BMC through the network interface from a remote machine. Use `ipmitool` for host and remote access.

- LAN interface—`ipmitool` is the standard tool to access BMC over the network. A dummy static IP address is preprogrammed in the BMC. You can change this dummy static IP address of the network interface using `ipmitool` from the microprocessor console:
  - `# ipmitool lan set 1 ipaddr <x.x.x.x>`

3

Configuration methods

The diagnostic operating software (DIAG OS) running on the local processor has `ipmitool` installed by default. You can use the `ipmitool` both at the switch and remotely.

Accessing BMC from the host does not require user name or password. The general syntax for using `ipmitool` is:

NOTE: -I and -H are optional.

```
ipmitool [-c|-h|-v|-V] -I lanplus -H <hostname> [-p <port>]
[-U <username>]
[-L <privlvl>]
[-a|-E|-P|-f <password>]
[-o <oemtype>]
[-O <sel oem>]
[-C <ciphersuite>]
[-Y|[-K|-k <kg_key>]
[-y <hex_kg_key>]
[-e <esc_char>]
[-N <sec>]
[-R <count>]
< command>
```

For example, to list sensors from the host use the following command from the host:

```
root@dellemc-diag-os:~# ipmitool sensor
PT_Mid_temp      | 31.000     | degrees C | ok    | na        | na        | na    | 78.000 | 80.000 | 85.000
NPU_Near_temp    | 29.000     | degrees C | ok    | na        | na        | na    | na     | na     | na
PT_Left_temp     | 28.000     | degrees C | ok    | na        | na        | na    | na     | na     | na
PT_Right_temp    | 30.000     | degrees C | ok    | na        | na        | na    | na     | na     | na
ILET_AF_temp     | 26.000     | degrees C | ok    | na        | na        | na    | na     | na     | na
PSU1_AF_temp     | 24.000     | degrees C | ok    | na        | na        | na    | 61.000 | 64.000 | na
PSU2_AF_temp     | 25.000     | degrees C | ok    | na        | na        | na    | na     | na     | na
PSU1_temp        | 34.000     | degrees C | ok    | na        | na        | na    | na     | na     | na
PSU2_temp        | na         | degrees C | na    | na        | na        | na    | na     | na     | na
CPU_temp         | 31.000     | degrees C | ok    | na        | na        | na    | 90.000 | 94.000 | na
FAN1_Rear_rpm    | 9120.000   | RPM       | ok    | na        | 1080.000  | na    | na     | na     | na
FAN2_Rear_rpm    | 9000.000   | RPM       | ok    | na        | 1080.000  | na    | na     | na     | na
FAN3_Rear_rpm    | 9000.000   | RPM       | ok    | na        | 1080.000  | na    | na     | na     | na
FAN4_Rear_rpm    | 9120.000   | RPM       | ok    | na        | 1080.000  | na    | na     | na     | na
FAN1_Front_rpm   | 10080.000  | RPM       | ok    | na        | 1080.000  | na    | na     | na     | na
FAN2_Front_rpm   | 10080.000  | RPM       | ok    | na        | 1080.000  | na    | na     | na     | na
FAN3_Front_rpm   | 9960.000   | RPM       | ok    | na        | 1080.000  | na    | na     | na     | na
FAN4_Front_rpm   | 10080.000  | RPM       | ok    | na        | 1080.000  | na    | na     | na     | na
PSU1_rpm         | 9000.000   | RPM       | ok    | na        | na        | na    | na     | na     | na
PSU2_rpm         | na         | RPM       | na    | na        | na        | na    | na     | na     | na
PSU_Total_watt   | 110.000    | Watts     | ok    | na        | na        | na    | na     | na     | na
PSU1_stat        | 0x0        | discrete  | 0x0180| na        | na        | na    | na     | na     | na
PSU2_stat        | 0x0        | discrete  | 0x0380| na        | na        | na    | na     | na     | na
PSU1_In_watt     | 110.000    | Watts     | ok    | na        | na        | na    | na     | na     | na
PSU1_In_volt     | 205.700    | Volts     | ok    | na        | na        | na    | na     | na     | na
PSU1_In_amp      | 0.480      | Amps      | ok    | na        | na        | na    | na     | na     | na
PSU1_Out_watt    | 90.000     | Watts     | ok    | na        | na        | na    | na     | na     | na
PSU1_Out_volt    | 12.400     | Volts     | ok    | na        | na        | na    | na     | na     | na
PSU1_Out_amp     | 7.500      | Amps      | ok    | na        | na        | na    | na     | na     | na
PSU2_In_watt     | na         | Watts     | na    | na        | na        | na    | na     | na     | na
PSU2_In_volt     | na         | Volts     | na    | na        | na        | na    | na     | na     | na
PSU2_In_amp      | na         | Amps      | na    | na        | na        | na    | na     | na     | na
PSU2_Out_watt    | na         | Watts     | na    | na        | na        | na    | na     | na     | na
PSU2_Out_volt    | na         | Volts     | na    | na        | na        | na    | na     | na     | na
PSU2_Out_amp     | na         | Amps      | na    | na        | na        | na    | na     | na     | na
ACPI_stat        | 0x0        | discrete  | 0x0180| na        | na        | na    | na     | na     | na
FAN1_prsnt       | 0x0        | discrete  | 0x0180| na        | na        | na    | na     | na     | na
FAN2_prsnt       | 0x0        | discrete  | 0x0180| na        | na        | na    | na     | na     | na
FAN3_prsnt       | 0x0        | discrete  | 0x0180| na        | na        | na    | na     | na     | na
FAN4_prsnt       | 0x0        | discrete  | 0x0180| na        | na        | na    | na     | na     | na
FAN1_Rear_stat   | 0x0        | discrete  | 0x0080| na        | na        | na    | na     | na     | na
FAN2_Rear_stat   | 0x0        | discrete  | 0x0080| na        | na        | na    | na     | na     | na
FAN3_Rear_stat   | 0x0        | discrete  | 0x0080| na        | na        | na    | na     | na     | na
FAN4_Rear_stat   | 0x0        | discrete  | 0x0080| na        | na        | na    | na     | na     | na
FAN1_Front_stat  | 0x0        | discrete  | 0x0080| na        | na        | na    | na     | na     | na
FAN2_Front_stat  | 0x0        | discrete  | 0x0080| na        | na        | na    | na     | na     | na
FAN3_Front_stat  | 0x0        | discrete  | 0x0080| na        | na        | na    | na     | na     | na
FAN4_Front_stat  | 0x0        | discrete  | 0x0080| na        | na        | na    | na     | na     | na
```

```
INTER_5.0V_volt | 4.900      | Volts      | ok    | 4.200     | 4.500     | 4.700 | 5.200  | 5.500  | 5.700
INTER_3.3V_volt | 3.300      | Volts      | ok    | 2.800     | 3.000     | 3.100 | 3.500  | 3.600  | 3.800
FPGA_1.0V_volt  | 0.990      | Volts      | ok    | 0.850     | 0.900     | 0.950 | 1.050  | 1.100  | 1.150
FPGA_1.2V_volt  | 1.190      | Volts      | ok    | 1.020     | 1.080     | 1.140 | 1.260  | 1.320  | 1.380
FPGA_1.8V_volt  | 1.780      | Volts      | ok    | 1.530     | 1.620     | 1.710 | 1.890  | 1.980  | 2.070
FPGA_3.3V_volt  | 3.200      | Volts      | ok    | 2.800     | 3.000     | 3.100 | 3.500  | 3.600  | 3.800
BMC_2.5V_volt   | 2.400      | Volts      | ok    | 2.100     | 2.200     | 2.300 | 2.600  | 2.800  | 2.900
BMC_1.15V_volt  | 1.150      | Volts      | ok    | 0.980     | 1.030     | 1.090 | 1.210  | 1.270  | 1.320
BMC_1.2V_volt   | 1.210      | Volts      | ok    | 1.020     | 1.080     | 1.140 | 1.260  | 1.320  | 1.380
SWITCH_6.8V_volt| 7.000      | Volts      | ok    | 5.800     | 6.100     | 6.400 | 7.200  | 7.500  | 7.800
SWITCH_3.3V_volt| 3.300      | Volts      | ok    | 2.800     | 3.000     | 3.100 | 3.500  | 3.600  | 3.800
SWITCH_1.8V_volt| 1.790      | Volts      | ok    | 1.530     | 1.620     | 1.710 | 1.890  | 1.980  | 2.070
USB_5.0V_volt   | 4.900      | Volts      | ok    | 4.200     | 4.500     | 4.700 | 5.200  | 5.500  | 5.700
NPU_1.2V_volt   | 1.190      | Volts      | ok    | 1.020     | 1.080     | 1.140 | 1.260  | 1.320  | 1.380
NPU_VDDCORE_volt| 0.800      | Volts      | ok    | 0.700     | 0.720     | 0.740 | 0.910  | 0.930  | 0.950
NPU_VDDANLG_volt| 0.790      | Volts      | ok    | 0.680     | 0.720     | 0.760 | 0.840  | 0.880  | 0.920
BMC boot        | 0x0        | discrete   | 0x0180| na        | na        | na    | na     | na     | na
SEL_sensor      | 0x0        | discrete   | 0x1080| na        | na        | na    | na     | na     | na
```

The command parameters change slightly when using `ipmitool` over LAN:

```
root@dellemc-diag-os:~# ipmitool -U admin -P admin -I lanplus -H 10.11.227.105 sensor
PT_Mid_temp     | 32.000     | degrees C  | ok    | na        | na        | na    | 78.000 | 80.000 | 85.000
NPU_Near_temp   | 29.000     | degrees C  | ok    | na        | na        | na    | na     | na     | na
PT_Left_temp    | 28.000     | degrees C  | ok    | na        | na        | na    | na     | na     | na
PT_Right_temp   | 30.000     | degrees C  | ok    | na        | na        | na    | na     | na     | na
ILET_AF_temp    | 26.000     | degrees C  | ok    | na        | na        | na    | na     | na     | na
PSU1_AF_temp    | 24.000     | degrees C  | ok    | na        | na        | na    | 61.000 | 64.000 | na
PSU2_AF_temp    | 25.000     | degrees C  | ok    | na        | na        | na    | na     | na     | na
PSU1_temp       | 33.000     | degrees C  | ok    | na        | na        | na    | na     | na     | na
PSU2_temp       | na         | degrees C  | na    | na        | na        | na    | na     | na     | na
CPU_temp        | 31.000     | degrees C  | ok    | na        | na        | na    | 90.000 | 94.000 | na
FAN1_Rear_rpm   | 9120.000   | RPM        | ok    | na        | 1080.000  | na    | na     | na     | na
FAN2_Rear_rpm   | 9000.000   | RPM        | ok    | na        | 1080.000  | na    | na     | na     | na
FAN3_Rear_rpm   | 9000.000   | RPM        | ok    | na        | 1080.000  | na    | na     | na     | na
FAN4_Rear_rpm   | 9000.000   | RPM        | ok    | na        | 1080.000  | na    | na     | na     | na
FAN1_Front_rpm  | 10080.000  | RPM        | ok    | na        | 1080.000  | na    | na     | na     | na
FAN2_Front_rpm  | 10080.000  | RPM        | ok    | na        | 1080.000  | na    | na     | na     | na
FAN3_Front_rpm  | 10080.000  | RPM        | ok    | na        | 1080.000  | na    | na     | na     | na
FAN4_Front_rpm  | 10080.000  | RPM        | ok    | na        | 1080.000  | na    | na     | na     | na
PSU1_rpm        | 9120.000   | RPM        | ok    | na        | na        | na    | na     | na     | na
PSU2_rpm        | na         | RPM        | na    | na        | na        | na    | na     | na     | na
PSU_Total_watt  | 110.000    | Watts      | ok    | na        | na        | na    | na     | na     | na
PSU1_stat       | 0x0        | discrete   | 0x0180| na        | na        | na    | na     | na     | na
PSU2_stat       | 0x0        | discrete   | 0x0380| na        | na        | na    | na     | na     | na
PSU1_In_watt    | 110.000    | Watts      | ok    | na        | na        | na    | na     | na     | na
PSU1_In_volt    | 205.700    | Volts      | ok    | na        | na        | na    | na     | na     | na
PSU1_In_amp     | 0.480      | Amps       | ok    | na        | na        | na    | na     | na     | na
PSU1_Out_watt   | 90.000     | Watts      | ok    | na        | na        | na    | na     | na     | na
PSU1_Out_volt   | 12.400     | Volts      | ok    | na        | na        | na    | na     | na     | na
PSU1_Out_amp    | 7.500      | Amps       | ok    | na        | na        | na    | na     | na     | na
PSU2_In_watt    | na         | Watts      | na    | na        | na        | na    | na     | na     | na
PSU2_In_volt    | na         | Volts      | na    | na        | na        | na    | na     | na     | na
PSU2_In_amp     | na         | Amps       | na    | na        | na        | na    | na     | na     | na
PSU2_Out_watt   | na         | Watts      | na    | na        | na        | na    | na     | na     | na
PSU2_Out_volt   | na         | Volts      | na    | na        | na        | na    | na     | na     | na
PSU2_Out_amp    | na         | Amps       | na    | na        | na        | na    | na     | na     | na
ACPI_stat       | 0x0        | discrete   | 0x0180| na        | na        | na    | na     | na     | na
FAN1_prsnt      | 0x0        | discrete   | 0x0180| na        | na        | na    | na     | na     | na
FAN2_prsnt      | 0x0        | discrete   | 0x0180| na        | na        | na    | na     | na     | na
FAN3_prsnt      | 0x0        | discrete   | 0x0180| na        | na        | na    | na     | na     | na
FAN4_prsnt      | 0x0        | discrete   | 0x0180| na        | na        | na    | na     | na     | na
FAN1_Rear_stat  | 0x0        | discrete   | 0x0080| na        | na        | na    | na     | na     | na
FAN2_Rear_stat  | 0x0        | discrete   | 0x0080| na        | na        | na    | na     | na     | na
FAN3_Rear_stat  | 0x0        | discrete   | 0x0080| na        | na        | na    | na     | na     | na
FAN4_Rear_stat  | 0x0        | discrete   | 0x0080| na        | na        | na    | na     | na     | na
FAN1_Front_stat | 0x0        | discrete   | 0x0080| na        | na        | na    | na     | na     | na
FAN2_Front_stat | 0x0        | discrete   | 0x0080| na        | na        | na    | na     | na     | na
FAN3_Front_stat | 0x0        | discrete   | 0x0080| na        | na        | na    | na     | na     | na
FAN4_Front_stat | 0x0        | discrete   | 0x0080| na        | na        | na    | na     | na     | na
INTER_5.0V_volt | 4.900      | Volts      | ok    | 4.200     | 4.500     | 4.700 | 5.200  | 5.500  | 5.700
INTER_3.3V_volt | 3.300      | Volts      | ok    | 2.800     | 3.000     | 3.100 | 3.500  | 3.600  | 3.800
FPGA_1.0V_volt  | 0.990      | Volts      | ok    | 0.850     | 0.900     | 0.950 | 1.050  | 1.100  | 1.150
FPGA_1.2V_volt  | 1.190      | Volts      | ok    | 1.020     | 1.080     | 1.140 | 1.260  | 1.320  | 1.380
FPGA_1.8V_volt  | 1.780      | Volts      | ok    | 1.530     | 1.620     | 1.710 | 1.890  | 1.980  | 2.070
FPGA_3.3V_volt  | 3.200      | Volts      | ok    | 2.800     | 3.000     | 3.100 | 3.500  | 3.600  | 3.800
BMC_2.5V_volt   | 2.400      | Volts      | ok    | 2.100     | 2.200     | 2.300 | 2.600  | 2.800  | 2.900
BMC_1.15V_volt  | 1.150      | Volts      | ok    | 0.980     | 1.030     | 1.090 | 1.210  | 1.270  | 1.320
BMC_1.2V_volt   | 1.210      | Volts      | ok    | 1.020     | 1.080     | 1.140 | 1.260  | 1.320  | 1.380
SWITCH_6.8V_volt| 7.000      | Volts      | ok    | 5.800     | 6.100     | 6.400 | 7.200  | 7.500  | 7.800
SWITCH_3.3V_volt| 3.300      | Volts      | ok    | 2.800     | 3.000     | 3.100 | 3.500  | 3.600  | 3.800
SWITCH_1.8V_volt| 1.790      | Volts      | ok    | 1.530     | 1.620     | 1.710 | 1.890  | 1.980  | 2.070
USB_5.0V_volt   | 4.900      | Volts      | ok    | 4.200     | 4.500     | 4.700 | 5.200  | 5.500  | 5.700
NPU_1.2V_volt   | 1.190      | Volts      | ok    | 1.020     | 1.080     | 1.140 | 1.260  | 1.320  | 1.380
NPU_VDDCORE_volt| 0.800      | Volts      | ok    | 0.700     | 0.720     | 0.740 | 0.910  | 0.930  | 0.950
NPU_VDDANLG_volt| 0.790      | Volts      | ok    | 0.680     | 0.720     | 0.760 | 0.840  | 0.880  | 0.920
BMC boot        | 0x0        | discrete   | 0x0180| na        | na        | na    | na     | na     | na
SEL_sensor      | 0x0        | discrete   | 0x1080| na        | na        | na    | na     | na     | na
```

To access BMC over a LAN, use the following `ipmitool` command:

```
ipmitool [-c|-h|-v|-V] -I lanplus -H <hostname> [-p <port>]
[-U <username>]
[-L <privlvl>]
[-a|-E|-P|-f <password>]
[-o <oemtype>]
[-O <sel oem>]
[-C <ciphersuite>]
[-Y|[-K|- <kg_key>]
[-y <hex_kg_key>]
[-e <esc_char>]
[-N <sec>]
[-R <count>]
<command>
```

If needed, you can download `ipmitool` from the `https://sourceforge.net/ projects/ipmitool` website. The commands to install `ipmitool` on Ubuntu or Fedora versions are as follows:

1. Install `ipmitool` on Ubuntu versions.

   ```
   # apt-get install ipmitool
   ```

2. Install `ipmitool` on Fedora versions.

   ```
   # yum install ipmitool
   ```

Run standard IPMI commands from `ipmitool`. For the command format, see *Intelligent Platform Management Interface Specification Second Generation v2.0.pdf*. For more documentation, see *https://linux.die.net/man/1/ipmitool*.

NOTE: Throughout this user guide, *Intelligent Platform Management Interface Specification Second Generation v2.0.pdf* is known as *IPMI Specification v2.0*. For more information about IPMI, see the IPMI resources that is hosted by Intel at https://www.intel.com/content/www/us/en/servers/ipmi/ipmi-technical-resources.html.

Topics:

- Configurations
- Date and time
- SNMP and email alerts
- Add and delete users
- Firewall
- Event log
- Default configuration restore

Configurations

LAN configurations

For network settings, see the *IPMI Specification v2.0* chapter 23.1 *Set LAN Configuration Parameters Command* and Table 23-4 *LAN Configuration Parameters*.

In addition to setting IP addresses, use `ipmitool` to set the network mask, MAC address, default gateway IP and MAC addresses, and so forth.

`ipmitool` commands:

```
root@dellemc-diag-os:~#  ipmitool lan set 1

usage: lan set <channel> <command> <parameter>
LAN set command/parameter options:
ipaddr <x.x.x.x>            Set channel IP address
netmask <x.x.x.x>           Set channel IP netmask
macaddr <x:x:x:x:x:x>       Set channel MAC address
defgw ipaddr <x.x.x.x>      Set default gateway IP address
defgw macaddr <x:x:x:x:x:x> Set default gateway MAC address  bakgw
ipaddr <x.x.x.x>            Set backup gateway IP address
```

```
bakgw macaddr <x:x:x:x:x:x> Set backup gateway MAC address
password <password>         Set session password for this channel
snmp <community string>     Set SNMP public community string
user                        Enable default user for this channel
access <on|off>             Enable or disable access to this channel
alert <on|off>              Enable or disable PEF alerting for this channel
arp respond <on|off>        Enable or disable BMC ARP responding
arp generate <on|off>       Enable or disable BMC gratuitous ARP generation
arp interval <seconds>      Set gratuitous ARP generation interval
vlan id <off|<id>>          Disable or enable VLAN and set ID (1-4094)
vlan priority <priority>    Set vlan priority (0-7)
auth <level> <type,..>      Set channel authentication types
  level  = CALLBACK, USER, OPERATOR, ADMIN
  type   = NONE, MD2, MD5, PASSWORD, OEM
ipsrc <source>                  Set IP Address source
  none   = unspecified source
  static = address manually configured to be static
  dhcp   = address obtained by BMC running DHCP
  bios   = address loaded by BIOS or system software
cipher_privs XXXXXXXXXXXXXXX   Set RMCP+ cipher suite privilege levels
X = Cipher Suite Unused
c = CALLBACK
u = USER
o = OPERATOR
a = ADMIN
O = OEM  bad_pass_thresh <thresh_num> <1|0> <reset_interval> <lockout_interval>
                                Set bad password threshold
```

NOTE: Dell Technologies recommends setting LAN parameters from the host microprocessor. You can run all other `ipmitool` options from a remote machine after the BMC has the correct IP address and LAN settings. When running `ipmitool` from a remote machine, the command prefix is `ipmitool -H <ip address of BMC> -I lanplus -U <user_name> -P <password> …">`.

The *<channel>* number is the LAN channel, which is `1` in this BMC implementation.

Dell Technologies recommends using the LAN settings command from a system-side machine rather than from a remote machine. To set a dynamic host configuration protocol (DHCP) IP address, use the following command:

```
# ipmitool lan set 1 ipsrc dhcp
```

To set a static IP address:

```
# ipmitool lan set 1 ipsrc static
# ipmitool lan set 1 ipaddr <x.x.x.x>
```

You can also add the BMC IP address from the BIOS. For more information, see the BIOS manual at www.dell.com/support.

DNS configuration

Use these commands to set and get domain name server (DNS)-related settings, for example hostname, domain setting, and DNS server settings. BMC supports only three DNS server IP addresses. These IP addresses can be either IPv4 or IPv6.

To set DNS configuration details, use the DNS configuration command. The DNS configuration is buffered and applies only after you set a DNS Restart—`parameter #7`.

Date and time

BIOS sets the date and time during boot up. Use the `iseltime` tool that is part of the `ipmiutil` package. Use the `ipmiutil` command only on the local processor. For more information about the `ipmiutil` command, see ipmiutil package.

Install the `ipmiutil` package and use the `iseltime` command.

To override the date and time that is used in the system event log (SEL) log, use the following command:

```
root@dellemc-diag-os:~# ipmitool sel time get
08/01/2018 15:10:46
root@dellemc-diag-os:~# ipmitool sel time set
usage: sel time set "mm/dd/yyyy hh:mm:ss"
root@dellemc-diag-os:~#
```

For `ipmiutil/iseltime`, download and install the binaries and documentation from https://ipmiutil.sourceforge.net. Also, various Linux distributions have binary packages prebuilt and available for download.

SNMP and email alerts

Event filters

To set the platform event filters, use the `raw` command format. To configure an entry in the filter table:

```
root@dellemc-diag-os:~# ipmitool raw 0x04 0x12 0x6 0x2 0xc0 0x1 0x2 0x2 0xff 0xff 0xff 0xff 0xff 0x01 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0 0x0
Byte 3 (0x60) - event filter table cmd
Byte 4(0x2) - filter number
Byte 5(0xc0) - filter config(enable)
Byte 6(0x1) - action(alert)
Byte 7(0x2) - policy number
Byte 8(0x2)  - event severity(information)
Byte 9(0xff) - slave address
Byte 10 (0xff) - channel number(any)
Byte 11(0xff)  - sensor number(any)
Byte 12(0x01) - event trigger(threshold)
```

The entry 2 is changed after the command, as shown:

```
root@dellemc-diag-os:~#
root@dellemc-diag-os:~#  ipmitool pef filter list
1 | disabled, configurable
2 | enabled, pre-configured | Any | Any | Information | OEM | Any | Alert | 2
3 | disabled, configurable
4 | disabled, configurable
5 | disabled, configurable
6 | disabled, configurable
7 | disabled, configurable
8 | disabled, configurable
9 | disabled, configurable
10 | disabled, configurable
11 | disabled, configurable
12 | disabled, configurable
13 | disabled, configurable
14 | disabled, configurable
15 | disabled, configurable
16 | disabled, configurable
17 | disabled, configurable
18 | disabled, configurable
19 | disabled, configurable
20 | disabled, configurable
21 | disabled, configurable
22 | disabled, configurable
23 | disabled, configurable
24 | disabled, configurable
25 | disabled, configurable
26 | disabled, configurable
27 | disabled, configurable
28 | disabled, configurable
29 | disabled, configurable
30 | disabled, configurable
31 | disabled, configurable
32 | disabled, configurable
```

```
33 | disabled, configurable
34 | disabled, configurable
35 | disabled, configurable
36 | disabled, configurable
37 | disabled, configurable
38 | disabled, configurable
39 | disabled, configurable
40 | disabled, configurable
```

For more information, see the *IPMI Specification v2.0* chapter 17.7 *Event Filter Table* and chapter 30.3 *Set PEF Configuration Parameters Command*.

Alert policies and destinations

For more information, see the *IPMI Specification v2.0* chapter 17.11 *Alert Policy Table* and chapter 30.3 *Set PEF Configuration Parameters Command (parameter 9)*.

LAN destinations

BMC supports SNMP alert destinations. These are SNMP traps. When you set a LAN destination for alerts, the BMC sends an SNMP trap to the set a destination whenever BMC detects alert conditions. You can setup the SNMP management application on the destination to receive these SNMP traps; however, setting up the SNMP management station is beyond the scope of this document.

To view alert destinations, use the `ipmitool lan alert print` command.

```
root@dellemc-diag-os:~#  ipmitool lan alert print
Alert Destination       : 0
Alert Acknowledge       : Unacknowledged
Destination Type        : PET Trap
Retry Interval          : 0
Number of Retries       : 0
Alert Gateway           : Default
Alert IP Address        : 0.0.0.0
Alert MAC Address       : 00:00:00:00:00:00
Alert Destination       : 1
Alert Acknowledge       : Unacknowledged
Destination Type        : PET Trap
Retry Interval          : 0
Number of Retries       : 0
Alert Gateway           : Default
Alert IP Address        : 0.0.0.0
Alert MAC Address       : 00:00:00:00:00:00
Alert Destination       : 2
Alert Acknowledge       : Unacknowledged
Destination Type        : PET Trap
Retry Interval          : 0
Number of Retries       : 0
Alert Gateway           : Default
Alert IP Address        : 0.0.0.0
Alert MAC Address       : 00:00:00:00:00:00
.
.
.
Alert Destination       : 15
Alert Acknowledge       : Unacknowledged
Destination Type        : PET Trap
Retry Interval          : 0
Number of Retries       : 0
Alert Gateway           : Default
Alert IP Address        : 0.0.0.0
Alert MAC Address       : 00:00:00:00:00:00
```

You can configure up to 15 destinations. To configure destination 1 to send an alert to a machine with IP address 10.11.227.180:

```
root@dellemc-diag-os:~#  ipmitool lan alert set 1 1 ipaddr 10.11.227.105
Setting LAN Alert 1 IP Address to 10.11.227.105
```

The following output using the `ipmitool lan alert print` command shows the configuration was successful:

```
root@dellemc-diag-os:~#  ipmitool lan alert print 1 1
Alert Destination       : 1
Alert Acknowledge       : Unacknowledged
Destination Type        : PET Trap
Retry Interval          : 0
Number of Retries       : 0
Alert Gateway           : Default
Alert IP Address        : 10.11.227.105
Alert MAC Address       : 00:00:00:00:00:00
```

Alert policy setup

To setup the alert policy, you must use the `ipmitool raw` command.

To view the current policy table, use the `ipmitool pef policy list` command.

```
root@dellemc-diag-os:~#  ipmitool pef policy list
1  | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
2 | 2 | enabled | Match-always | 1 | 802.3 LAN | PET | AMI | 0 | 0 | 10.11.227.105 | 00:00:00:00:00:00
3  | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
4  | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
5  | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
6  | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
7  | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
8  | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
9  | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
10 | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
.
.
.
57 | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
58 | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
59 | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
60 | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
```

There are 60 entries available for a policy table. The following example shows setting a policy entry. For a detailed description of the table entries, see the *IPMI Specification v2.0 Alert policy table entry*.

```
root@dellemc-diag-os:~#  ipmitool raw 0x4 0x12 0x9 0x2 0x28 0x11 0x00

root@dellemc-diag-os:~#  ipmitool pef policy list
1  | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
2 | 2 | enabled | Match-always | 1 | 802.3 LAN | PET | AMI | 0 | 0 | 10.11.227.105 | 00:00:00:00:00:00
3  | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
4  | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
5  | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
6  | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
7  | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
8  | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
9  | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
10 | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
.
.
.
57 | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
58 | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
59 | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
60 | 0 | disabled | Match-always | 0 | IPMB (I2C) | 0
```

Add and delete users

The following describes adding and deleting users:

There are 10 entries for a user list.

1. Add a new user by modifying one of the empty entries in the user list using the following:

```
$ ./ipmitool -H xx.xx.xxx.xx -I lanplus -U admin -P admin user set name 3 <name>
$ ./ipmitool -H xx.xx.xxx.xx -I lanplus -U admin -P admin user set password 3
Password for user 3:
Password for user 3:
Set User Password command successful (user 3)
```

   Step 1 creates a user with no access.

2. Set the privilege level for the user in Step 1 using the following:

```
$ ./ipmitool -H xx.xx.xxx.xx -I lanplus -U admin -P admin user  priv 3
User Commands:
  summary      [<channel number>]
  list         [<channel number>]
  set name     <user id> <username>
  set password <user id> [<password> <16|20>]
  disable      <user id>
  enable       <user id>
  priv         <user id> <privilege level> [<channel number>]
      Privilege levels:
      * 0x1 - Callback
      * 0x2 - User
      * 0x3 - Operator
      * 0x4 - Administrator
      * 0x5 - OEM Proprietary
      * 0xF - No Access

   test         <user id> <16|20> [<password]>

$ ./ipmitool -H xx.xx.xxx.xx -I lanplus -U admin -P admin user priv 3 2
Set Privilege Level command successful (user 3)
$ ./ipmitool -H xx.xx.xxx.xx -I lanplus -U admin -P admin user list
ID  Name     Callin  Link Auth  IPMI Msg   Channel Priv Limit
1            false   false      true       ADMINISTRATOR
2   admin    true    true       true       ADMINISTRATOR
3   <name>   true    true       true       USER
4            true    false      false      NO ACCESS
5            true    false      false      NO ACCESS
6            true    false      false      NO ACCESS
7            true    false      false      NO ACCESS
8            true    false      false      NO ACCESS
9            true    false      false      NO ACCESS
10           true    false      false      NO ACCESS
```

   You can individually enable channels for a certain privilege level access. For example, to place the LAN channel accessible for "USER" level access, use the following:

```
$ ./ipmitool -H xx.xx.xxx.xxx -I lanplus -U admin -P admin channel setaccess 1 3 callin=off link=off ipmi=on
privilege=1
Set User Access (channel 1 id 3) successful.
$ ./ipmitool -H xx.xx.xxx.xxx -I lanplus -L USER -U <name> -P <name> fru
Get Device ID command failed: 0xd4 Insufficient privilege level
FRU Device Description : Builtin FRU Device (ID 0)
Get Device ID command failed: Insufficient privilege level
$ ./ipmitool -H xx.xx.xxx.xxx -I lanplus -U admin -P admin channel setaccess 1 3 callin=off link=off ipmi=on
privilege=2
Set User Access (channel 1 id 3) successful.
$ ./ipmitool -H xx.xx.xxx.xx -I lanplus -L USER -U <name> -P <name> fru
FRU Device Description : Builtin FRU Device (ID 0)
 Board Mfg Date        : Mon Feb 12 08:00:00 2018
 Board Mfg             : Dell
 Board Product         : <platform>
 Board Serial          : CNCES0082C0002
 Board Part Number     : 0G1T60X01
 Product Manufacturer  : Dell
 Product Name          : <platform>
 Product Version       : 00
 Product Serial        : X1
```

```
 Product Asset Tag     : D4SSG02

FRU Device Description : FRU_PSU1 (ID 1)
 Unknown FRU header version 0x00

FRU Device Description : FRU_PSU2 (ID 2)
 Board Mfg Date        : Fri Jan 12 18:47:00 2018
 Board Mfg             : DELL
 Board Product         : PWR SPLY,495W,RDNT,DELTA
 Board Serial          : CNDED0081G01GL
 Board Part Number     : 0GRTNKA02

FRU Device Description : FRU_FAN1 (ID 3)
 Unknown FRU header version 0x00

FRU Device Description : FRU_FAN2 (ID 4)
 Board Mfg Date        : Mon Feb 12 08:01:00 2018
 Board Mfg             : Dell
 Board Product         : <platform>
 Board Serial          : CNCES008260036
 Board Part Number     : 07CRC9X01
 Product Manufacturer  : Dell
 Product Name          : <platform>
 Product Version       :
 Product Serial        :
 Product Asset Tag     : D4SSG02
```

For more information, see the *IPMI Specification v2.0* chapter 22.26 *Set User Access Command*, 22.28 *Set User Name Command*, and 22.30 *Set User Password Command*.

- Request data byte 1—[7]
  - 0b-Do not change the following bits in this byte
  - 1b-Enable changing bits in this byte
- Request data byte 1—[6] User restricted to callback
  - 0b-User Privilege Limit is determined by the User Privilege Limit parameter for both callback and non-callback connections.
  - 1b-User Privilege Limit is determined by the User Privilege Limit parameter for callback connections, but is restricted to Callback level for non-callback connections. A user can only initiate a callback when he/she 'calls in' to the BMC, but after the callback connect is made, the user could potentially establish a session as an Operator.
- Request data byte 1—[5] User link authentication enable/disable. This is used to enable/disable a user's name and password information for link authentication. Link authentication itself is a global setting for the channel and is enabled/disabled via the serial or moden configuration parameters.
  - 0b-disable user for link authentication
  - 1b-enable user for link authentication
- Request data byte 1—User IPMI Messaging enable/disable. This is used to enable/disable a user's name and password information for IPMI messaging. In this case, *IPMI Messaging* means the ability to execute generic IPMI commands that are not associated with a particular payload type. For example, if you disable IPMI Messaging for a user, but that user is enabled for activating the SOL payload type, IPMI commands associated with SOL and session management, such as *Get SOL Configuration parameters* and *Close Session* are available, but generic IPMI commadns such as *Get SEL Time* are not.
  - 0b-disable user for link authentication
  - 1b-enable user for link authentication
- Request data byte 2—User ID
  - [7:6] reserved
  - [5:0] User ID. 00000b = reserved
- Request data byte 3—User limits
  - [7:6] reserved
  - [3:0] User Privilege Limit. This determines the maximum privilege level that the user can to switch to on the specified channel.
    - 0h-reserved
    - 1h-Callback
    - 2h-User
    - 3h-Operator
    - 4h-Adminstrator
    - 5h-OEM Proprietary

    - Fh-NO ACCESS
- Request data byte (4)—User Session Limit. Optional—Sets how many simultaneous sessions are activated with the username associated with the user. If not supported, the username activates as many simultaneous sessions as the implementation supports. If an attempt is made to set a non-zero value, a CCh "invalid data field" error returns.
    - [7:4]-Reserved
    - [3:0]-User simultaneous session limit. 1=based. oh=only limited by the implementations support for simultaneous sessions.
- Response data byte 1—Completion code

  NOTE: If the user access level is set higher than the privilege limit for a given channel, the implementation does not return an error completion code. If required, It is up to the software to check the channel privilege limits set using the `Set Channel Access` command and provide notification of any mismatch.

Set User Name Command

- Request date byte 1—User ID
    - [7:6]-reserved
    - [5:0]-User ID. 000000b-reserved. User ID 1 is permanently associated with User 1, the null user name.
- Request date byte 2:17—User Name String in ASCII, 16 bytes maximum. Strings with fewer then 16 characters terminate with a null (00h) character. The 00h character is padded to 16 bytes. When the string is read back using the `Get User Name` command, those bytes return as 0s.
- Response data byte 1—Completion code

Set User Password Command

- Request data byte 1—User ID. For IPMI v20, the BMC supports 20-byte passwords (keys) for all user IDs that have configurable passwords. The BMC maintains an internal tag indicating if the password is set as a 16-byte or 20-byte password.

  Use a 16-byte password in algorithms that require a 20-byte password. The 16-byte password is padded with 0s to create 20-bytes.

  If an attempt is made to test a password that is stored as a 20-byte password as a 16-byte password, and vice versa, the `test password` operation returns a `test failed` error completion code.

  You cannot use a password stored as a 20-byte password to establish an IPMI v1.5 session. You must set the password as a 16-byte password to configure the same password for both IPMI v20 and IPMI v1.5 access. The password is padded with 0s as necessary.

  Use the test password operation to determine if a password is stored as 16-bytes or 20-bytes.

- Request data byte 2—
    - [7:2] Reserved
    - [1:0] Operation
        - 00b-disable user
        - 01b-enable user-10b-set password
        - 11b-test password. This compares the password data give in the request with the presently stored password and returns an OK completion code if it matches. Otherwise, an error completion code returns.
- Request data byte 3:18—For 16-byte passwords. Password data. This is a fixed-length required filed used for setting and testing password operations. If the user enters the password as an ASCII string, it must be null (00h) terminated 00h padded if the string is shorter than 16 bytes. This field is not needed for the `disable user` or `enable user` operation. If the field is present, the BMC ignores the data.
- Request data byte 3:22—For 20-byte passwords. This is a fixed-length required filed used for setting and testing password operations. If the user enters the password as an ASCII string, it must be null (00h) terminated 00h padded if the string is shorter than 20 bytes. This field is not needed for the `disable user` or `enable user` operation. If the field is present, the BMC ignores the data.
- Response data byte 1—Completion code. Generic plus the following command-specific completion codes:
    - 80h-mandatory password test failed. Password size is correct but the password data does not match the stored value.
    - 81h-mandatory password test failed. Wrong password size.

Firewall

To set a firewall, use the `set firewall configuration` command. Use parameters 0–3 to add the iptables rules and 4–7 to remove the iptables rules.

- NetFN—0x32
- Command—0x76
- Request data Byte 1—parameter selector
- Request data Byte 2—State selector
- Request data Byte 3:N—Configuration parameter data
- Response data Byte 1—Completion code
  - 80h—Parameter not supported
  - 81h—Invalid time (start/stop time)
  - 82h—Attempt to write read-only parameter
  - 83h—Attempt to access HTTP Port 80

To set the firewall configuration state, use the following:

Table 2. Firewall set parameters

Type specific param	#	Parameter data
To set the command to DROP	00	Parameter to drop packets. Parameter 0–3 uses this state to add the rules to drop the packets based on the IP address/port number or ange of IP addresses/port numbers. Use parameter 4–7 to remove the rule.
To set the command to ACCEPT	01	Parameter to accept packets. Parameter 0–3 uses this state to add the rules to accept the packets based on the IP address/port number or ange of IP addresses/port numbers. Use parameter 4–7 to remove the rule.

To set the firewall parameters, use the following:

Table 3. Firewall parameters

Parameter	#	Parameter data
Add the IPv4 address rule	0	Data 1:4—IP address • MS-byte first. This is an IPv4 address that is blocked or unblocked based on the state.
Add the range of IPv4 addresses rule	1	Data 1:8—IP address range • [1:4]—Starting IP address from which IPs are blocked or unblocked based on the state. • [5:8]—Ending IP address until IPs are blocked or unblocked based on the state. For example, if the IP address is x1.x2.x3.x4, the format is: • 1st byte = x1 • 2nd byte = x2 • 3rd byte = x3 • 4th byte = x4
Add the IPv4 port number rule	2	Data 1:—Protocol TCP/UDP • 0 = TCP

Table 3. Firewall parameters (continued)

Parameter	#	Parameter data
		• 1 = UDP • 2 = both TCP and UDP • Data 2:3—port number • [2:3]—MX byte first. Port number blocked or unblocked based on the state.
Add the Pv4 port number range rule	3	Data 1:—Protocol TCP/UDP • 0 = TCP • 1 = UDP • 2 = both TCP and UDP • Data 2:5—port range • [2:3]—Port number from the ports blocked or unblocked based on the state. • [4:5]—Port number till ports are blocked or unblocked based on the state.
Remove the IPv4 address rule	4	Data 1:4—IP address • MS-byte first. This is the IPv4 address type that is blocked or unblocked based on state.
Remove the range of IPv4 addresses rule	5	Data 1:8—IP address range • [1:4]—Starting IP address that is blocked or unblocked based on the state. • [5:8]—Ending IP address that is blocked or unblocked based on the state. For example, if the IP address is x1.x2.x3.x4, the format is: • 1st byte = x1 • 2nd byte = x2 • 3rd byte = x3 • 4th byte = x4
Remove the IPv4 port number rule	6	Data 1:—Protocol TCP/UDP • 0 = TCP • 1 = UDP • 2 = both TCP and UDP • Data 2:3—port number • [2:3]—Port number from the ports blocked or unblocked based on the state.
Remove the IPv4 port range rule	7	Data 1:—Protocol TCP and UDP • 0 = TCP • 1 = UDP • 2 = both TCP and UDP • Data 2:5—port range • [2:3]—Port number from the ports blocked or unblocked based on the state. • [4:5]—Port number till ports are blocked or unblocked based on the state.

Table 3. Firewall parameters (continued)

Parameter	#	Parameter data
Flush IPv4 and IPv6 iptable	8	Flush all the rules set using iptables and ip6tables.
Drop all	9	Add iptables rules to block IPv4 and IPv6 traffic to the BMC. The state selector is not used. • Data1: Protocol • Bit 7:2—Reserved • Bit 1—IPv6 • Bit 0—IPv4
Remove drop all rule	10	Remove iptables rules to block IPv4 and IPv6 traffic to the BMC. The state selector is not used. • Data1: Protocol • Bit 7:2—Reserved • Bit 1—IPv6 • Bit 0—IPv4
Add IPv4 address with timeout rule	11	Data 1:4—IP address • MS-byte first. The IPv4 address type blocked or unblocked based on the state. • Date 5:10—Start time • [5:6]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 7—month • 8—date • 9—hour • 10—minute • Date 11-16—stop time • [11:12]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 13—month • 14—date • 15—hour • 16—minute
Add IPv4 range of addresses with timeout rule	12	Data 1:8—IP address • [1:4]—Starting IP address blocked or unblocked based on the state. • [5:8]—Ending IP address till IPs are blocked or unblocked based on the state. • Date 9:14—Start time • [9:10]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 11—month • 12—date • 13—hour • 14—minute • Date 15-20—Stop time • [15:16]—Year

Table 3. Firewall parameters (continued)

Parameter	#	Parameter data
		• LS-byte first if little endian system. Two-byte data required to form year.
Add the IPv4 port number with timeout rule	13	Data 1—Protocol TCP and UDP • 0 = TCP • 1 = UDP • 2 = both TCP and UDP • Data 2:3—port number • [2:3]—Port number from the ports blocked or unblocked based on the state. • Date 4:9—Start time • [4:5]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 6—month • 7—date • 8—hour • 9—minute • Date 10-15—stop time • [10:11]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 12—month • 13—date • 14—hour • 15—minute
Add the IPv4 port range with timeout rule	14	Data 1:—Protocol TCP and UPD • 0 = TCP • 1 = UDP • 2 = both TCP and UDP • Data 2:5—port number • [2:3]—Port number from the ports blocked or unblocked based on the state. • [4:5]—Port number till the ports blocked or unblocked based on the state. • Date 6:11Start time • [6:7]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 8—month • 9—date • 10—hour • 11—minute • Date 12-17—stop time • [12:13]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 14—month • 15—date • 16—hour • 17—minute

Table 3. Firewall parameters (continued)

Parameter	#	Parameter data
Remove the IPv4 address with timeout rule	15	Data 1:4—IP address • MS-byte first. The IPv4 address type blocked or unblocked based on the state. • Date 5:10—Start time • [5:6]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 7—month • 8—date • 9—hour • 10—minute • Date 11-16—stop time • [11:12]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 13—month • 14—date • 15—hour • 16—minute
Remove the range IPv4 address with timeout rule	16	Data 1:8—IP address • [1:4]—Starting IP address blocked or unblocked based on the state. • [5:8]—Ending IP address till IPs are blocked or unblocked based on the state. • Date 9:14—Start time • [9:10]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 11—month • 12—date • 13—hour • 14—minute • Date 15-20—Stop time • [15:16]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 17—month • 18—date • 19—hour • 20—minute
Remove the IPv4 port number with timeout rule	17	Data 1—Protocol TCP and UDP • 0 = TCP • 1 = UDP • 2 = both TCP and UDP • Data 2:3—port number • [2:3]—Port number from the ports blocked or unblocked based on the state. • Date 4:9—Start time • [4:5]—Year • LS-byte first if little endian system. Two-byte data required to form year.

Table 3. Firewall parameters (continued)

Parameter	#	Parameter data
		• 6—month • 7—date • 8—hour • 9—minute • Date 10-15—stop time • [10:11]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 12—month • 13—date • 14—hour • 15—minute
Remove the IPv4 port number range with timeout rule	18	Data 1:—Protocol TCP and UPD • 0 = TCP • 1 = UDP • 2 = both TCP and UDP • Data 2:5—port number • [2:3]—Port number from the ports blocked or unblocked based on the state. • [4:5]—Port number till the ports blocked or unblocked based on the state. • Date 6:11Start time • [6:7]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 8—month • 9—date • 10—hour • 11—minute • Date 12-17—stop time • [12:13]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 14—month • 15—date • 16—hour • 17—minute
Drop all IPv4 or IPv6 with timeout rule	19	Add iptables rules to block IPv4 and IPv6 traffic to the BMC. The state selector is not used. • Data1: Protocol • Bit 7:2—Reserved • Bit 1—IPv6 • Bit 0—IPv4 • Date 2:7—Start time • [2:3]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 4—month • 5—date • 6—hour

Table 3. Firewall parameters (continued)

Parameter	#	Parameter data
		• 7—minute • Date 8:13—Stop time • [8:9]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 10—month • 11—date • 12—hour • 13—minute
Remove drop all Ipv4 or IPv6 with timeout rule	20	Add iptables rules to block IPv4 and IPv6 traffic to the BMC. The state selector is not used. • Data1: Protocol • Bit 7:2—Reserved • Bit 1—IPv6 • Bit 0—IPv4 • Date 2:7—Start time • [2:3]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 4—month • 5—date • 6—hour • 7—minute • Date 8:13—Stop time • [8:9]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 10—month • 11—date • 12—hour • 13—minute
Add IPv6 address with timeout rule	21	Data 1:16—IPv6 address • MS-byte first. The IPv6 address type blocked or unblocked based on the state. • Date 7:22—Start time • [17:18]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 19—month • 20—date • 21—hour • 22—minute • Date 23-28—stop time • [23:24]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 25—month • 26—date • 27—hour • 28—minute

Table 3. Firewall parameters (continued)

Parameter	#	Parameter data
Add IPv6 address range with timeout rule	22	Data 1:16—IPv6 address range • [1:16]—Port number from the ports blocked or unblocked based on the state. • [17:32]—Port number till the ports blocked or unblocked based on the state. • Date 33:38—Start time • [33:34]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 35—month • 36—date • 37—hour • 38—minute • Date 39:44—stop time • [39:40]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 41—month • 42—date • 43—hour • 44—minute
Remove the IPv6 address with timeout rule	23	Data 1:16—IPv6 address • MS-byte first. The IPv4 address type blocked or unblocked based on the state. • Date 17:22—Start time • [17:18]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 19—month • 20—date • 21—hour • 22—minute • Date 23-28—stop time • [23:24]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 25—month • 26—date • 27—hour • 28—minute
Remove the Ipv6 address range with timeout rule	24	Data 1:16—IPv6 address range • [1:16]—Port number from the ports blocked or unblocked based on the state. • [17:32]—Port number till the ports blocked or unblocked based on the state. • Date 33:38—Start time • [33:34]—Year • LS-byte first if little endian system. Two-byte data required to form year.

Table 3. Firewall parameters (continued)

Parameter	#	Parameter data
		• 35—month • 36—date • 37—hour • 38—minute • Date 39:44—stop time • [39:40]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 41—month • 42—date • 43—hour • 44—minute
Add the IPv6 port number with timeout rule	25	Data 1—Protocol TCP and UDP • 0 = TCP • 1 = UDP • 2 = both TCP and UDP • Data 2:3—port number • [2:3]—Port number from the ports blocked or unblocked based on the state. • Date 4:9—Start time • [4:5]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 6—month • 7—date • 8—hour • 9—minute • Date 10-15—stop time • [10:11]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 12—month • 13—date • 14—hour • 15—minute
Add the IPv6 port number range with timeout rule	26	Data 1—Protocol TCP and UDP • 0 = TCP • 1 = UDP • 2 = both TCP and UDP • Data 2:5—port number • [2:3]—Port number from the ports blocked or unblocked based on the state. • [4:5]—Year • Date 6:11—Start time • [6:7]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 8—month • 9—date • 10—hour • 11—minute

Table 3. Firewall parameters (continued)

Parameter	#	Parameter data
		• Date 12-17—stop time • [12:13]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 14—month • 15—date • 16—hour • 17—minute
Remove the IPv6 port number with timeout rule	27	Data 1—Protocol TCP and UDP • 0 = TCP • 1 = UDP • 2 = both TCP and UDP • Data 2:3—port number • [2:3]—Port number from the ports blocked or unblocked based on the state. • [4:9]—Year • Date 4:9—Start time • [4:5]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 6—month • 7—date • 8—hour • 9—minute • Date 10-15—stop time • [10:11]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 12—month • 12—date • 14—hour • 15—minute
Remove the IPv6 port range with timeout rule	28	Data 1—Protocol TCP and UDP • 0 = TCP • 1 = UDP • 2 = both TCP and UDP • Data 2:5—port number • [2:3]—Port number from the ports blocked or unblocked based on the state. • [4:5]—Year • Date 6:11—Start time • [6:7]—Year • LS-byte first if little endian system. Two-byte data required to form year. • 8—month • 9—date • 10—hour • 11—minute • Date 12-17—stop time • [12:13]—Year

Table 3. Firewall parameters (continued)

Parameter	#	Parameter data
		• LS-byte first if little endian system. Two-byte data required to form year. • 14—month • 15—date • 16—hour • 17—minute
Add the IPv6 address rule	29	Data 1:16—IPv6 address. • MS-byte first. This is an IPv6 address that is blocked or unblocked based on state.
Add the IPv6 address range rule	30	Data 1:16—IPv6 address range • [1:16]—Starting IP address from which IPs are blocked or unblocked based on the state. • [17.32]—Ending IP address until IPs are blocked or unblocked based on the state.
Remove the IPv6 address rule	31	Data 1:16—IPv6 address • MS-byte first. This is an IPv6 address that is blocked or unblocked based on state.
Remove the IPv6 address range rule	32	Data 1:16—IPv6 address range • [1:16]—Starting IP address from which IPs are blocked or unblocked based on the state. • [17.32]—Ending IP address until IPs are blocked or unblocked based on the state.
Add the IPv6 port number rule	33	Data 1—Protocol TCP and UDP • 0 = TCP • 1 = UDP • 2 = both TCP and UDP • Data 2:3—port number • [2:3]—Port number from the ports blocked or unblocked based on the state.
Add the IPv6 port number range rule	34	Data 1—Protocol TCP and UDP • 0 = TCP • 1 = UDP • 2 = both TCP and UDP • Data 2:5—port number • [2:3]—Port number from the ports blocked or unblocked based on the state. • [4:5]—Port number till the ports are blocked or u nblocked based on the state.
Remove the IPv6 port number rule	35	Data 1—Protocol TCP and UDP • 0 = TCP • 1 = UDP • 2 = both TCP and UDP • Data 2:3—port number

Table 3. Firewall parameters (continued)

Parameter	#	Parameter data
		• [2:3]—Port number from the ports blocked or unblocked based on the state.
Remove the IPv6 port number range rule	36	Data 1—Protocol TCP and UDP • 0 = TCP • 1 = UDP • 2 = both TCP and UDP • Data 2:5—port number • [2:3]—Port number from the ports blocked or unblocked based on the state. • [4:5]—Port number till the ports are blocked or u nblocked based on the state.

Event log

To get the IPMI event log, use the `ipmitool sel list` command.

To clear the event log, use the `ipmitool sel clear` command.

For IPMI event log settings, see the *IPMI Specification v2.0* chapter 31.4 *Reserve SEL Command* and 31.5 *Get SEL Entry Command*.

Reserve system event log (SEL) command

Use reserve SEL to set the present owner of the SEL. This reservation provides a limited amount of protection on repository access from the IPMB when you delete or incrementally read records. Use get SEL to read the SEL repository.

- Response data byte 1—Completion code
  - 81h—cannot execute the command, SEL erase in progress
- Response data byte 2—Reservation ID, LS byte 0000h reserved.
- Response data byte 3—Reservation ID, SM byte

Get SEL command

- Request data byte 1:2—Reservation IS, LS byte first. Only required for a partial get. Otherwise use 0000h.
- Request data byte 3:4—SEL record ID, LS byte first.
  - 0000h=GET FIRST ENTRY
  - FFFFh=GET LAST ENTRY
- Request data byte 5—Offset into record
- Request data byte 6—Bytes to read. FFH means read entire record.
- Response data byte 1—Completion code. Returns an error completion code if the SEL is empty.
  - 81h=cannot execute the command, SEL erase in progress.
- Response data byte 2:3—Next SEL record ID. LS byte first (returns FFFFh if the record just returned is the last record).
  - **NOTE:** FFFFh is not allowed as the record ID of an actual record. For example, the record ID in Record Data for the last record cannot be FFFFh.
- Response data byte 4:N—Record data, 16 bytes for the entire record.

Set LOG configuration command

To set the system or audit log configuration, use the `set LOG configuration` command.

- Netfn—0x32
- Command—0x68

Audit log configuration

- Request data byte 1—Cmd
  - [7:2] Reserved
  - [1:0] 01h–Audit log
- Request data byte 1—Status
  - [7:2] Reserved
  - [1:0] 01h–Disabled
  - 01h–Enable local
- Response data byte 1—00h-success
  - CCh=invalid data field
  - FFh=unspecified error
- Response data byte 1—Cmd
  - [7:2] Reserved
  - [1:0] 00h–system log
- Response data byte 2—Status
  - [7:2] Reserved
  - [1:0] 01h–Disabled
  - 01h–Enable local
- Response data byte 3-70 for REMOTE (68 bytes) or 3-7 for LOCAL (5 bytes)—ENABLED REMOTE
  - **NOTE:** These request data bytes are required only when you enable either the local or remote system log.

```
64bytes : Hostname (ASCII)
Remote syslog server
4bytes : port number
```

To set the remote server ip address to 10.0.124.22 and port to 770:

```
ipmitool -I lanplus -H xx.xx.xx.xx -U xxx -P xxx raw 0x32 0x68 0x00
0x02 0x31 0x30 0x2e 0x30 0x2e 0x31 0x32 0x34 0x2e 0x32 0x32 0x00
0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00
0x02 0x03 0x00 0x00
ENABLED LOCAL
4bytes : Size (LSB first)
size of each file to rotate (file size is from 3 to 65535 KB)
1bytes : Rotate
Number of back-up files after logrotate (maximum 1 file)
```

To set the file size to 100 bytes, use the IPMI command:

```
ipmitool -I lanplus -H xx.xx.xx.xx -U xxx -P xxx raw 0x32 0x68 0x00
0x01 0x64 0x00 0x00 0x00 0x01
```

Default configuration restore

Use configuration restore to start the configuration from scratch. For example, use configuration restore to remove the old configuration and start over if you reinstall the system or move the system to a new location.

Restore default configuration command

- NetFn—0x32

- Command—0x66
- Response byte 1—Completion code

Default settings

The following tables list the default settings after a switch restore.

Table 4. Default settings after a switch restore

Name	Setting
BMC OOB	Enabled for non-TAA and disabled for TAA
BMC OOB — after restore to default	Disabled
BMC WEB	Enabled for non-TAA and disabled for TAA
BMC WEB — after restore to default	Disabled
BMC console	Enabled for non-TAA and disabled for TAA
BMC console — after restore to default	Enabled for non-TAA and disabled for TAA
BMC supports unique password	Yes
BMC OOB username/password	admin/admin
BMC OOB username/password — after restore to default	admin/**admin** (but only valid for the IPMI commands for `mc info` and change administrator password)
BMC WEB	admin/admin
BMC WEB — after restore to default	admin/**admin** (but WEB GUI displays a message to confirm change of the administrator password)
BMC console login username/password	sysadmin/superuser
BMC console login username/password — after restore to default	sysadmin/superuser

Set backup configuration flag

To set the backup flags for the `manage BMC confirguration` command, use the `set backup configuration flag` command.

- NetFN—0x32
- Command—0xF3
- Request data byte 1:2—Byte 1 is the value specifies to back up a configuration feature or not.
  - [7]—Reserved
  - [6]—1b: Backup SNMP. 0b: Do not backup the simple network management protocol (SNMP)
  - [5]—1b: Backup SYSLOC. 0b: Do not backup SYSLOG
  - [4]—1b: Backup KVM. 0b: Do not backup keyboard, video, and mouse (KVM)
  - [3]—1b: Backup NTP. 0b: Do not backup network time protocol (NTP)
  - [2]—1b: Backup IPMI. 0b: Do not backup IPMI
  - [1]—1b: Backup NETWORK And SERVICES. 0b: Do not backup NETWORK And SERVICES
  - [0]—1b: Backup AUTHENTICATION. 0b: Do not backup AUTHENTICATION

  NOTE: Reserved bits may be updated further based on the requirement.
- Response data byte 1—Completion code
  - 0x83—Authentication feature is not enabled
  - 0x84—NTP feature is not enabled
  - 0x85—KVM feature is not enabled
  - 0x86—SNMP feature is not enabled

4

Host power control

The following are host power control commands:

- `Power Off`—the `ipmitool` powers off
- `Power On`—the `ipmitool` powers on
- `Power Cycle`—the `ipmitool` power cycles
- `Hard Reset`—the `ipmitool` power resets

5

Firmware update

To update the firmware from a remote machine, use the BMC LAN interface.

You can also update the firmware in the local host OS using the USB interface. The USB interface is between the BMC and the microprocessor. When using the USB, the BMC simulates a virtual USB device, then Yafuflash sends the image to the BMC via the USB bus. Typically the update process completes in five minutes.

For more information about Yafuflash, see the *S5200-ON Series Release Notes*.

Table 5. Firmware update

Tool	Medium	OS	Comments
Yafuflash	USB	Linux	Recommended—Host OS only
Yafuflash	LAN	WIndows/Linux	Internal use only

The BMC virtual USB is disabled by default. Enable the USB before you update the firmware.

Update BMC by USB interface

Enable BMC virtual USB:

```
ipmitool raw 0x32 0xaa 0x00 (Then wait 15s)
```

Update Main BMC:

```
./Yafuflash -cd -mse 1 rom.ima
```

Update BMC by LAN interface

1. Ensure that the client Linux or Windows machine can ping the BMC IP address.
2. Open a command window.
3. Update the main BMC using the following command:

```
./Yafuflash -nw -ip bmc_ip -u admin -p admin -mse 1 bmc.ima
```

6

Access system health sensors

To check sensor information, use the following command:

```
root@dellemc-diag-os:~# ipmitool sensor list
```

To change the sensor threshold, see the *IPMI Specification v2.0* chapter 35.8 *Set Sensor Thresholds Command*.

- Request data byte 1—Sensor number, FFH=reserved
- Request data byte 2—
  - [7:6] - reserved. Write as 00b
  - [5] - 1b=set upper non-recoverable threshold
  - [4] - 1b=set upper critical threshold
  - [3] - 1b=set upper non-critical threshold
  - [2] - 1b=set lower non-recoverable threshold
  - [1] - 1b=set lower critical threshold
  - [0] - 1b=set lower non-critical threshold
- Request data byte 3—lower non-critical threshold. Ignored if bit 0 of byte 2 = 0
- Request data byte 4—lower critical threshold. Ignored if bit 1 of byte 2 = 0
- Request data byte 5—lower non-recoverable threshold. Ignored if bit 2 of byte 2 = 0
- Request data byte 6—upper non-critical threshold. Ignored if bit 3 of byte 2 = 0
- Request data byte 7—upper critical threshold value. Ignored if bit 4 of byte 2 = 0
- Request data byte 8—upper non-recoverable threshold value. Ignored if bit 5 of byte 2 = 0
- Response data byte 1—Completion code

ipmitool sensors

```
root@dellemc-diag-os:~# ipmitool sensor list
PT_Mid_temp      | 32.000     | degrees C | ok    | na    | na        | na    | 78.000  | 80.000  | 85.000
NPU_Near_temp    | 29.000     | degrees C | ok    | na    | na        | na    | na      | na      | na
PT_Left_temp     | 28.000     | degrees C | ok    | na    | na        | na    | na      | na      | na
PT_Right_temp    | 30.000     | degrees C | ok    | na    | na        | na    | na      | na      | na
ILET_AF_temp     | 26.000     | degrees C | ok    | na    | na        | na    | na      | na      | na
PSU1_AF_temp     | 24.000     | degrees C | ok    | na    | na        | na    | 61.000  | 64.000  | na
PSU2_AF_temp     | 25.000     | degrees C | ok    | na    | na        | na    | na      | na      | na
PSU1_temp        | 33.000     | degrees C | ok    | na    | na        | na    | na      | na      | na
PSU2_temp        | na         | degrees C | na    | na    | na        | na    | na      | na      | na
CPU_temp         | 31.000     | degrees C | ok    | na    | na        | na    | 90.000  | 94.000  | na
FAN1_Rear_rpm    | 9120.000   | RPM       | ok    | na    | 1080.000  | na    | na      | na      | na
FAN2_Rear_rpm    | 9000.000   | RPM       | ok    | na    | 1080.000  | na    | na      | na      | na
FAN3_Rear_rpm    | 9000.000   | RPM       | ok    | na    | 1080.000  | na    | na      | na      | na
FAN4_Rear_rpm    | 9000.000   | RPM       | ok    | na    | 1080.000  | na    | na      | na      | na
FAN1_Front_rpm   | 10080.000  | RPM       | ok    | na    | 1080.000  | na    | na      | na      | na
FAN2_Front_rpm   | 10080.000  | RPM       | ok    | na    | 1080.000  | na    | na      | na      | na
FAN3_Front_rpm   | 10080.000  | RPM       | ok    | na    | 1080.000  | na    | na      | na      | na
FAN4_Front_rpm   | 10080.000  | RPM       | ok    | na    | 1080.000  | na    | na      | na      | na
PSU1_rpm         | 9000.000   | RPM       | ok    | na    | na        | na    | na      | na      | na
PSU2_rpm         | na         | RPM       | na    | na    | na        | na    | na      | na      | na
PSU_Total_watt   | 110.000    | Watts     | ok    | na    | na        | na    | na      | na      | na
PSU1_stat        | 0x0        | discrete  | 0x0180| na    | na        | na    | na      | na      | na
PSU2_stat        | 0x0        | discrete  | 0x0380| na    | na        | na    | na      | na      | na
PSU1_In_watt     | 110.000    | Watts     | ok    | na    | na        | na    | na      | na      | na
PSU1_In_volt     | 205.700    | Volts     | ok    | na    | na        | na    | na      | na      | na
PSU1_In_amp      | 0.480      | Amps      | ok    | na    | na        | na    | na      | na      | na
PSU1_Out_watt    | 90.000     | Watts     | ok    | na    | na        | na    | na      | na      | na
PSU1_Out_volt    | 12.400     | Volts     | ok    | na    | na        | na    | na      | na      | na
PSU1_Out_amp     | 7.500      | Amps      | ok    | na    | na        | na    | na      | na      | na
PSU2_In_watt     | na         | Watts     | na    | na    | na        | na    | na      | na      | na
PSU2_In_volt     | na         | Volts     | na    | na    | na        | na    | na      | na      | na
PSU2_In_amp      | na         | Amps      | na    | na    | na        | na    | na      | na      | na
PSU2_Out_watt    | na         | Watts     | na    | na    | na        | na    | na      | na      | na
PSU2_Out_volt    | na         | Volts     | na    | na    | na        | na    | na      | na      | na
PSU2_Out_amp     | na         | Amps      | na    | na    | na        | na    | na      | na      | na
ACPI_stat        | 0x0        | discrete  | 0x0180| na    | na        | na    | na      | na      | na
FAN1_prsnt       | 0x0        | discrete  | 0x0180| na    | na        | na    | na      | na      | na
FAN2_prsnt       | 0x0        | discrete  | 0x0180| na    | na        | na    | na      | na      | na
```

```
FAN3_prsnt       | 0x0       | discrete  | 0x0180| na    | na       | na    | na     | na     | na
FAN4_prsnt       | 0x0       | discrete  | 0x0180| na    | na       | na    | na     | na     | na
FAN1_Rear_stat   | 0x0       | discrete  | 0x0080| na    | na       | na    | na     | na     | na
FAN2_Rear_stat   | 0x0       | discrete  | 0x0080| na    | na       | na    | na     | na     | na
FAN3_Rear_stat   | 0x0       | discrete  | 0x0080| na    | na       | na    | na     | na     | na
FAN4_Rear_stat   | 0x0       | discrete  | 0x0080| na    | na       | na    | na     | na     | na
FAN1_Front_stat  | 0x0       | discrete  | 0x0080| na    | na       | na    | na     | na     | na
FAN2_Front_stat  | 0x0       | discrete  | 0x0080| na    | na       | na    | na     | na     | na
FAN3_Front_stat  | 0x0       | discrete  | 0x0080| na    | na       | na    | na     | na     | na
FAN4_Front_stat  | 0x0       | discrete  | 0x0080| na    | na       | na    | na     | na     | na
INTER_5.0V_volt  | 4.900     | Volts     | ok    | 4.200 | 4.500    | 4.700 | 5.200  | 5.500  | 5.700
INTER_3.3V_volt  | 3.300     | Volts     | ok    | 2.800 | 3.000    | 3.100 | 3.500  | 3.600  | 3.800
FPGA_1.0V_volt   | 0.990     | Volts     | ok    | 0.850 | 0.900    | 0.950 | 1.050  | 1.100  | 1.150
FPGA_1.2V_volt   | 1.190     | Volts     | ok    | 1.020 | 1.080    | 1.140 | 1.260  | 1.320  | 1.380
FPGA_1.8V_volt   | 1.780     | Volts     | ok    | 1.530 | 1.620    | 1.710 | 1.890  | 1.980  | 2.070
FPGA_3.3V_volt   | 3.200     | Volts     | ok    | 2.800 | 3.000    | 3.100 | 3.500  | 3.600  | 3.800
BMC_2.5V_volt    | 2.400     | Volts     | ok    | 2.100 | 2.200    | 2.300 | 2.600  | 2.800  | 2.900
BMC_1.15V_volt   | 1.150     | Volts     | ok    | 0.980 | 1.030    | 1.090 | 1.210  | 1.270  | 1.320
BMC_1.2V_volt    | 1.210     | Volts     | ok    | 1.020 | 1.080    | 1.140 | 1.260  | 1.320  | 1.380
SWITCH_6.8V_volt| 7.000     | Volts     | ok    | 5.800 | 6.100    | 6.400 | 7.200  | 7.500  | 7.800
SWITCH_3.3V_volt| 3.300     | Volts     | ok    | 2.800 | 3.000    | 3.100 | 3.500  | 3.600  | 3.800
SWITCH_1.8V_volt| 1.790     | Volts     | ok    | 1.530 | 1.620    | 1.710 | 1.890  | 1.980  | 2.070
USB_5.0V_volt    | 4.900     | Volts     | ok    | 4.200 | 4.500    | 4.700 | 5.200  | 5.500  | 5.700
NPU_1.2V_volt    | 1.190     | Volts     | ok    | 1.020 | 1.080    | 1.140 | 1.260  | 1.320  | 1.380
NPU_VDDCORE_volt| 0.800     | Volts     | ok    | 0.700 | 0.720    | 0.740 | 0.910  | 0.930  | 0.950
NPU_VDDANLG_volt| 0.790     | Volts     | ok    | 0.680 | 0.720    | 0.760 | 0.840  | 0.880  | 0.920
BMC boot         | 0x0       | discrete  | 0x0180| na    | na       | na    | na     | na     | na
SEL_sensor       | 0x0       | discrete  | 0x1080| na    | na       | na    | na     | na     | na
```

7

IPMI commands

NOTE: All commands are subject to change as the `ipmi` commands evolve over time.

- `ipmi raw`
- `ipmi i2c`
- `ipmi ian print`
- `ipmi ian set`
- `ipmi ian alert`
- `ipmi chassis status`
- `ipmi chassis selftest`
- `ipmi chassis power status`
- `ipmi chassis power up / on`
- `ipmi chassis power down / off`
- `ipmi chassis power cycle`
- `ipmi chassis identify`
- `ipmi chassis poh`
- `ipmi chassis restart_cause`
- `ipmi chassis policy list`
- `ipmi chassis policy always-on`
- `ipmi chassis policy previous`
- `ipmi chassis policy always-off`
- `ipmi chassis bootparam get <param #>`
- `ipmi chassis bootparam set bootparam set bootflag <device>`
  - **Legal devices are:**
  - `none` : No override
  - `force_pxe` : Force PXE boot
  - `force_disk` : Force boot from default hard-drive
  - `force_safe` : Force boot from default hard-drive, request Safe Mode
  - `force_diag` : Force boot from diagnostic partition
  - `force_cdrom` : Force boot from CD/DVD
  - `force_bios` : Force boot into BIOS setup
  - **Legal options are:**
  - `help` : Print this message
  - `PEF` : Clear valid bit on reset/power cycle caused by PEF
  - `timeout` : Automatically clear boot flag valid bit on timeout
  - `watchdog`: Clear valid bit on reset/power cycle caused by watchdog
  - `reset` : Clear valid bit on push button reset/soft reset
  - `power` : Clear valid bit on power up via power push button or wake event
  - Any Option may be prepended with `no-` to invert sense of operation
- `ipmi chassis bootdev <device> bios`
- `ipmi event <num>`
- `ipmi event file <filename>`
- `ipmi event event<sensorid><state> [event_dir]`
- `ipmi mc reset <warm | cold>`
- `ipmi mc guid`
- `ipmi mc info`
- `ipmi mc watchdog get`
- `ipmi mc watchdog reset`

- ipmi mc watchdog off
- ipmi mc selftest
- ipmi mc getenables
- ipmi mc getenabled *<item><option=on | off>*
- ipmi mc getsysinfo *<argument>* system_fw_version
- ipmi mc getsysinfo *<argument>* primary_os_name
- ipmi mc getsysinfo *<argument>* os_name
- ipmi mc getsysinfo *<argument>* system_nam
- ipmi mc setsysinfo *<argument>* system_fw_version
- ipmi mc setsysinfo *<argument>* primary_os_name
- ipmi mc setsysinfo *<argument>* os_name
- ipmi mc setsysinfo *<argument>* system_nam
- ipmi sdr list | elist [option] all
- ipmi sdr list | elist [option] full
- ipmi sdr list | elist [option] compact
- ipmi sdr list | elist [option] event
- ipmi sdr list | elist [option] mcloc
- ipmi sdr list | elist [option] fru
- ipmi sdr list | elist [option] generic
- ipmi sdr type [option] *<Senfor_Type>*
- ipmi sdr type [option] list
- ipmi sdr get *<Sensor_ID>*
- ipmi sdr info
- ipmi sdr entity *<Entity_ID>*[.*<Instance_ID>*]
- ipmi sdr dump *<file>*
- ipmi sensor list
- ipmi sensor thresh *<id><threshold><setting>*
- ipmi sensor get *<id>*
- ipmi sensor reading *<id>*
- ipmi fru print [fru id]
- ipmi fru read *<fru id><fru file>*
- ipmi fru write *<fru id><fru file>*
- ipmi fru fru internaluse
- ipmi sel info
- ipmi sel clear
- ipmi sel delete *<id>*
- ipmi sel list
- ipmi sel elist
- ipmi sel get
- ipmi sel add *<filename>*
- ipmi sel time get
- ipmi sel time set
- ipmi sel save *<filename>*
- ipmi sel redraw *<filename>*
- ipmi sel writeraw *<filename>*
- ipmi pef info
- ipmi pef status
- ipmi pef policy list
- ipmi pef policy enable
- ipmi pef policy disable
- ipmi pef policy create
- ipmi pef policy delete
- ipmi sol info [*<channel number>*]
- ipmi sol set *<parameter><value>*[channel]

- ipmi sol payload <*enable*|*disable*|*status*>[channel][userid]
- ipmi sol activate [<*usesolkeepalive*|*n)eepalive*>][instance=<*number*>]
- ipmi sol deactivate [instance=<*number*>]
- ipmi sol looptest [<*loop times*>[<*loop interval(in ms)*>[<*instance*>]]]
- ipmi user summary [<*channel number*>]
- ipmi user list [<*channel number*>]
- ipmi user set name <*user id*><*username*>
- ipmi user set password <*user id*>[<*password*><*16*|*20*>]
- ipmi user disable <*user id*>
- ipmi user enable <*user id*>
- ipmi user priv <*user id*><*privilege level*>[<*channel number*>]
- ipmi user test <*user id*><*16*|*20*>[<*password*>]
- ipmi channel authcap <*channel number*><*max privilege*>
- ipmi channel getaccess <*channel number*>[user id]
- ipmi channel setaccess <*channel number*><*user id*>[callin=on][ipmi=on|off][link=on] [privilege=level]
- ipmi channel info [channel number]
- ipmi channel getciphers <*ipmi* | *sol*>[channel]
- ipmi session info <*active* | *all* | *id 0xnnnnnnn* | *handle 0xnn*>
- ipmi dcmi discover
- ipmi dcmi power<*command*> reading
- ipmi dcmi power<*command*> get_limit
- ipmi dcmi power<*command*> set_limit
- ipmi dcmi power<*command*> activate
- ipmi dcmi power<*command*> deactivate
- ipmi dcmi sensors
- ipmi dcmi asset_tag
- ipmi dcmi set_asset_tag
- ipmi dcmi get_mc_id_string
- ipmi dcmi set_mc_id_string
- ipmi dcmi get_temp_reading
- ipmi dcmi get_conf_param
- ipmi dcmi set_conf_param
- ipmi dcmi oob_discover
- ipmi shell
- ipmi exec
- ipmi set
  - **Options are:**
  - hostname <host> : Session hostname
  - username <user> : Session username
  - password <pass> : Session password
  - privlvl <level> : Session privilege level force
  - authtype <type> : Authentication type force
  - localaddr <addr> : Local IPMB address
  - targetaddr <addr> : Remote target IPMB address
  - port <port> : Remote RMCP port
  - csv [level] : Enable output in comma-separated format
  - verbose [level] : Verbose level

8

ipmiutil package

NOTE: All commands are subject to change as the `ipmiutil` package evolves over time. For more information about the IPMI utility, use cases, and the newest list of subcommands, see the IPMI website that is hosted by Intel at https://www.intel.com/content/www/us/en/servers/ipmi/ipmi-technical-resources.html.

- `ipmiutil`—a metacommand to invoke each of the following functions:
  - `ipmiutil alarms (ialarms)`—show and set the front panel alarms, including light emitting diodes (LEDs) and relays.
  - `ipmiutil config (iconfig)`—list, save, or restore the BMC configuration parameters.
  - `ipmiutil cmd (icmd)`—send specific IPMI commands to the BMC for testing and debug purposes.
  - `ipmiutil discover (idiscover)`—discover the available IPMI LAN nodes on a subnet.
  - `ipmiutil events (ievents)`—a stand-alone utility to decode IPMI events and platform event trap (PET) data.
  - `ipmiutil firewall (ifirewall)`—discover the available IPMI LAN nodes on a subnet.
  - `ipmiutil fru (ifru)`—show decoded field replaceable units (FRU) board/product inventory data and write FRU asset tags.
  - `ifruset`—show decoded FRU inventory data and set a FRU product area.
  - `iseltime`—show and set the IPMI system event log (SEL) time according to the system time.
  - `ipmiutil fwum (ifwum)`—OEM firmware update manager extensions
  - `ipmiutil getevt (igetevent)`—receive any IPMI events and display them.
  - `ipmiutil health (ihealth)`—check and report the basic health of the IPMI BMC.
  - `ipmiutil hpm (ihpm)`—hardware platform management (HPM) firmware update manager extensions
  - `ipmiutil lan (ilan)`—show and configure the local area network (LAN) port and platform event filter (PEF) table to generate BMC LAN alerts using the firmware events.
  - `ipmiutil picmg (ipicmg)`—discover the available IPMI LAN nodes on a subnet.
  - `ipmiutil reset (ireset)`—cause the BMC to hard reset or power down the system.
  - `ipmiutil sel (isel)`—a tool to show the firmware system event log (SEL) records.
  - `ipmiutil sensor (isensor)`—show the sensor data records (SDR), readings, and thresholds.
  - `ipmiutil serial (iserial)`—a tool to show and configure the BMC serial port for various modes, for example, Terminal mode.
  - `ipmiutil sol (isol)`—start or stop an IPMI serial-over-LAN console session.
  - `ipmiutil sunoem (isunoem)`—Sun OEM functions.
  - `ipmiutil wdt (iwdt)`—show and set the watchdog timer.
  - `checksel`—cron script using `impiutil sel` to check the SEL, write new events to the OS system log, and clear the SEL if nearly full.
  - `ipmi_port`—daemon to bind the remote management control protocol (RMCP) port and sleep to prevent Linux portmap from stealing the RMCP port.
  - `ipmi_wdt`—initial script to restart the watchdog timer every 60 seconds using the cron.
  - `ipmi_asy`—initial script that runs the `ipmiutil getevt -a` command for a remote shutdown.
  - `ipmi_evt`—initial script the runs the `imput getevt -s` command for monitoring events.
  - `hpiutil/*`—parallel hardware platform interface (HPI) utilities that conform to the SA Forum Hardware Platform Interface. Also a basis of the `openhpi/clients/`
  - `bmc_panic`—a kernel patch to save information if the system panics. The command is found in the OpenIPMI driver in kernels 2.6 and greater and in the Intel IMB driver in version 28 and greater

9

Access FRU data

To check field replacement unit (FRU) data, use the following command:

```
root@dellemc-diag-os:~# ipmitool fru print
```

For more FRU information, see the *IPMI Specification v2.0* chapter 34.2 *Read FRU Data Command*.

- Request data 1—FRU device ID. FFh=reserved
- Request data 2—FRU inventory offset to read, LS byte
- Request data 3—FRU inventory offset to read, LS byte
  - Offset is in bytes or words-per-device. Access type returned in the `Get FRU Inventory Area Info` command output.
- Request data 4—Count to read. Count is '1' based.
- Response data 1—Completion code. Generic, plus the command specifics:
  - 81h=FRU device busy. The requested cannot be completed because the logical FRU device is in a state where FRU information is temporarily unavailable. This state is possibly due to a loss of arbitration if the FRU implements as a device on a shared bus.
  - Software can elect to retry the operation after a minimum of 30 milliseconds if the code returns. Dell Technologies recommends that the management controllers incorporate built-in retry mechanisms. Generic IPMI does not take advantage of this completion code.
- Response data 2—Count returned. Count is '1' based.
- Response data 3:2=N—Requested data

ipmitool FRUs

```
root@dellemc-diag-os:~# ipmitool fru print
FRU Device Description : Builtin FRU Device (ID 0)
Board Mfg Date         : Sat May 19 06:04:00 2018
Board Mfg              : CES00
Board Product          : <platform>
Board Serial           : CN01XR4WCES0085F0002
Board Part Number      : 01XR4WX01
Product Manufacturer   : CES00
Product Name           : <platform>
Product Asset Tag      : GDNRG02
FRU Device Description : PSU1_fru (ID 1)
Board Mfg Date         : Fri Mar 30 21:30:00 2018
Board Mfg              : DELL
Board Product          : PWR SPLY,750W,AC,PS/IO,DELTA
Board Serial           : CNDED0083U00D5
Board Part Number      : 0HXWNFA00FRU
Device Description : PSU2_fru (ID 2)
Board Mfg Date         : Fri Mar 30 22:12:00 2018
Board Mfg              : DELL
Board Product          : PWR SPLY,750W,AC,PS/IO,DELTA
Board Serial           : CNDED0083U00BY
Board Part Number      : 0HXWNFA00FRU
Device Description : FAN1_fru (ID 3)
Board Mfg Date         : Mon Jan  1 00:00:00 1996
Board Serial           : CN07R5RFCES0084N0081
Board Part Number      : 07R5RFX01FRU
Device Description : FAN2_fru (ID 4)
Board Mfg Date         : Mon Jan  1 00:00:00 1996
Board Serial           : CN07R5RFCES0084N0080
Board Part Number      : 07R5RFX01FRU
Device Description : FAN3_fru (ID 5)
Board Mfg Date         : Mon Jan  1 00:00:00 1996
Board Serial           : CN07R5RFCES0084N0083
```

```
Board Part Number      : 07R5RFX01FRU
Device Description : FAN4_fru (ID 6)
Board Mfg Date         : Mon Jan  1 00:00:00 1996
Board Serial           : CN07R5RFCES0084N0082
Board Part Number      : 07R5RFX01
```

10

Dell EMC support

The Dell EMC support site provides documents and tools to help you use Dell EMC equipment and mitigate network outages. Through the support site you can obtain technical information, access software upgrades and patches, download available management software, and manage your open cases. The Dell EMC support site provides integrated, secure access to these services.

To access the Dell EMC support site, go to www.dell.com/support/. To display information in your language, scroll down to the bottom of the web page and select your country or region from the drop-down menu.

- To obtain product-specific information, enter the 7-character service tag, which is known as a luggage tag, or 11-digit express service code of your switch and click **Submit**.
- To view the platform service tag or express service code, pull out the luggage tag on the upper-right side of the platform or retrieve it remotely using the `ipmitool -H <bmc ip address> -I lanplus -U <user name> -P <password> fru` command.
- To receive more technical support, click **Contact Us**. On the Contact Information web page, click **Technical Support**.

To access switch documentation, go to www.dell.com/support/ and enter your switch type.

To search for drivers and downloads, go to **Drivers & Downloads** tab for your switch.

To participate in Dell EMC community blogs and forums, go to www.dell.com/community.