DM-SV01 Server

BMC User Manual

Revision 3.0 – Last updated Jun 2023

DATACOM ELECTRONICS GMBH

The information and specifications provided in this manual are subject to change without notice and are not recognized as any kind of contract. Datacom specifically disclaims any warranties, expressed or implied, of merchantability, fitness for any particular purpose and noninfringement, including those arising from a course of usage, dealing or trade practice.

Although every precaution has been taken in the preparation of this document, Datacom does not assume any liability for any errors or omissions as well as no obligation is assumed for damage resulting from the use of the information contained in this manual.

In no event will Datacom or its suppliers be liable for any direct, indirect, special, speculative, consequential or incidental, including, without limitation, lost profits or loss or damage to data or hardware arising out of the use or inability to use this product or this manual, even in case Datacom or its suppliers have been advised of the possibility of such damages. In particular, Datacom shall not have liability for the costs of replacing, repairing or recovering software, data or hardware related to the product or its use.

Table of Contents

1 Introduction

The BMC (Baseboard Management Controller) is a specialized microprocessor present in the DM-SV01 mainboard and responsible for a set of monitoring and management functions of the host system. The BMC concept is widely employed in the datacenter industry and this solution is present in a wide range of products, such as servers, storage systems, switches, etc. The main advantage of having a BMC in the product is that it allows the system management staff to remotely monitor and control several functions of the system, without the need of being physically present in the datacenter facilities.

Some of the main functions provided by the BMC are the following:

- Inventory data management for several system components (such as CPUs, memories, NICs, etc).
- Monitoring the system temperatures and controlling the cooling system (FANs) in order to provide a secure and efficient thermal behavior by means of a dedicated PID controller.
- Monitoring the server temperatures and power consumption for generating alarms whenever a critical threshold is exceeded.
- Control system power operations, such as CPUs reset and power on/off.
- Remote BIOS/UEFI and BMC FW update.
- KVM for remotely accessing the video output of the host processors, as well as to virtually interact with the server using the mouse and keyboard from your remote workstation.
- Virtual media, which allows the user to load an ISO file for OS installation or maintenance purposes.
- Serial over LAN (SoL), which allows the user to access the output console of the host system by means of the BMC web GUI.
- LDAP, which allows the authentication of users by means of an external LDAP server.

The BMC operation is independent of the host CPUs. This means that the BMC can be reset or have its FW upgraded without affecting the host CPUs processing functions. Also the CPUs can be reset and turned on/off without affecting the BMC management and monitoring functionalities.

In the DM-SV01, the BMC can be accessed and controlled by a dedicated Ethernet port present in the front panel of the system (out of band management) or by means of the Ethernet port from the mezzanine NIC card, sharing the traffic with the host CPUs (in band management). Details of the network connections for the BMC can be found in the section “2.4.1 Network settings”.

When remotely accessing the BMC, the server administrator has almost full control over the server without the need of being physically present in the datacenter premises. As an example, the user may load an ISO file for OS installation by means of the virtual media, turn the host processors on through the BMC and then use the KVM to interact with the BIOS/UEFI and the OS installer.

All these functions can be accessed remotely and are available in the BMC web management Graphical User Interface (GUI). Once the user has network connectivity to the BMC, the BMC web GUI can be accessed by means of the network browser from the remote workstation. Detailed information regarding the web GUI and all the functions available can be found in the section “2 BMC Web GUI”.

1.1 DM-SV01 BMC FW

The BMC FW for the DM-SV01 has been developed using the OpenBMC distribution as a starting point to implement the features and functionalities required to properly manage the server. OpenBMC is an open source FW distribution available at github (https://github.com/openbmc/openbmc), which uses some customizable components to allow developers to adopt it as a basis for developing its own BMC management solution.

The DM-SV01 BMC uses the OpenBMC distribution as the base FW platform, over which all the server's specific functionalities have been implemented, such as temperature and power sensors monitoring, FANs control, add-in cards management, FW upgrade policy, network settings customizations, etc.

The BMC is capable of upgrading its own firmware by means of the Web management GUI. The FW update does not affect the host CPUs, so they can keep running its workloads while the update is performed. Details about the FW upgrade process can be found in the section "2.4.2 Firmware".

The DM-SV01 also provides an additional read-only recovery BMC FW which is stored in a separate memory device. If the main FW for some reason is not capable of booting, the BMC read-only FW is loaded automatically and allows the user to access some basic functionalities of the BMC. Thus, the system manager does not lose access to the server and can use the BMC read-only FW to keep managing and monitoring the system while trying to recover the main BMC FW. When the read-only BMC FW is loaded, the user will be able to see a warning message in the web GUI, as shown in the Figure 1.

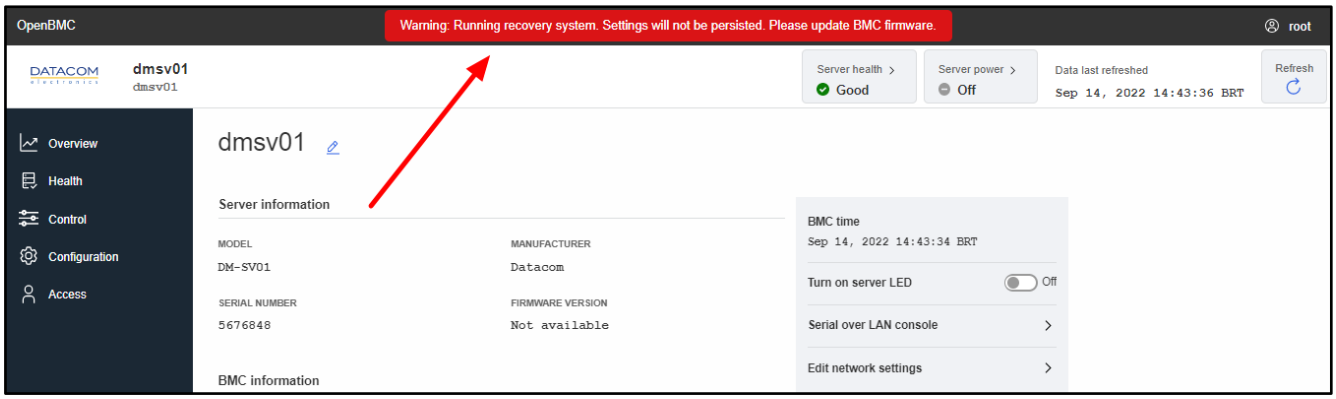


Figure 1: BMC web GUI - warning message for read-only FW

Important: when the BMC read-only FW is running, the settings will not be persistent. This means that any change performed at the BMC configurations will be lost in the first BMC reset. So, it is highly recommended to use the read-only boot only for recovering the main BMC FW.

1.2 Getting started

The access to the BMC is done by means of a network connection. By default, the Ethernet ports of the BMC are configured as DHCP clients. So, the user can connect to the BMC dedicated Ethernet port in the DM-SV01 front panel for an out of band management or to the Mezzanine NIC port 0 for an in band traffic, and activate a DHCP server to send an IP Address to the BMC. Detailed information about the BMC network configuration can be found in section "2.4.1 Network settings".

After that, it is possible to use your preferred network browser and enter with the BMC IP Address preceded by "https://" in the address bar. This will open the login web page of the BMC.

By default, the root user is available as a full access administrator of the system and can be logged in by using the credentials below:

- **Username:** root
- **Password:** 0penBmc (please note that the first character is the number "zero")

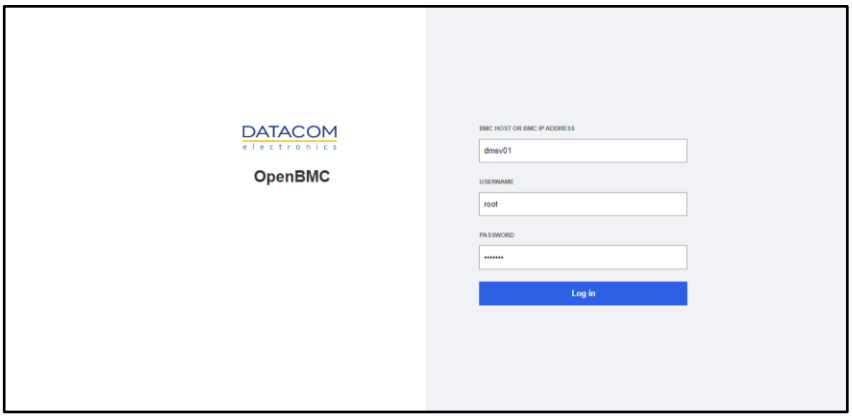


Figure 2: BMC Login prompt

After the login is performed, the user has full access to the BMC web GUI and can navigate through all the settings described in the section “2 BMC Web GUI”. As a first step, it is recommended to change the password of the root user as a security practice. The procedure for changing the password, as well as additional details regarding the users administration, is described in section “2.5.2.2 Managing users”.

1.3 BMC password recovery

If the user loses access to the BMC or forgets the login credentials, there is a procedure for providing the reset of the username and password. However, this procedure requires physical access to the server, as well as a complete reset of the BMC settings to the factory defaults.

The procedure for recovering the BMC login credentials is described below:

1. Completely power the server off, unplugging it from the DM1904 chassis or removing the 12V power from the server.
2. Press and hold the ID button at the server front panel while powering the server on.
3. Keep the ID button pressed until the ID LED blinks two times. Then release the ID button.

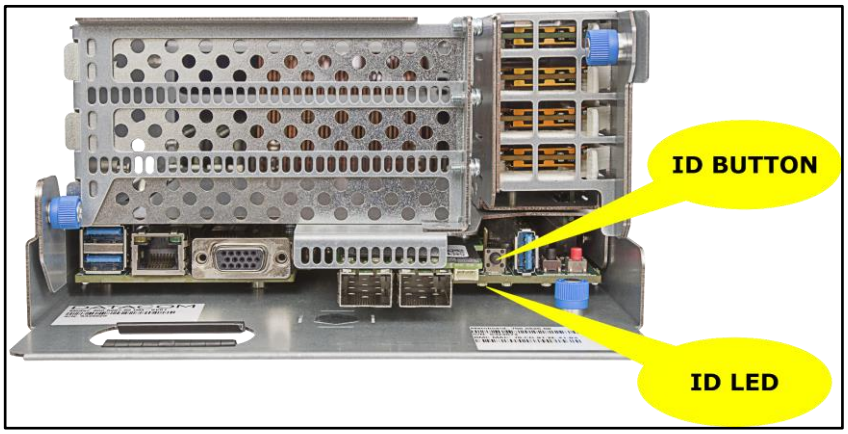


Figure 3: DM-SV01 ID button and ID LED

4. This procedure will force the BMC to boot its recovery FW.

a. The BMC recovery FW is very similar to the main FW. However, the recovery FW will always boot with factory settings and any changes performed in the BMC menus will not be persisted after a power cycle in the DM-SV01 server.

5. The BMC recovery FW can be accessed normally by following the procedure described in section “1.2 Getting started”. The user can confirm that the BMC is running the recovery FW by checking the text message in the red box as shown in the image below.

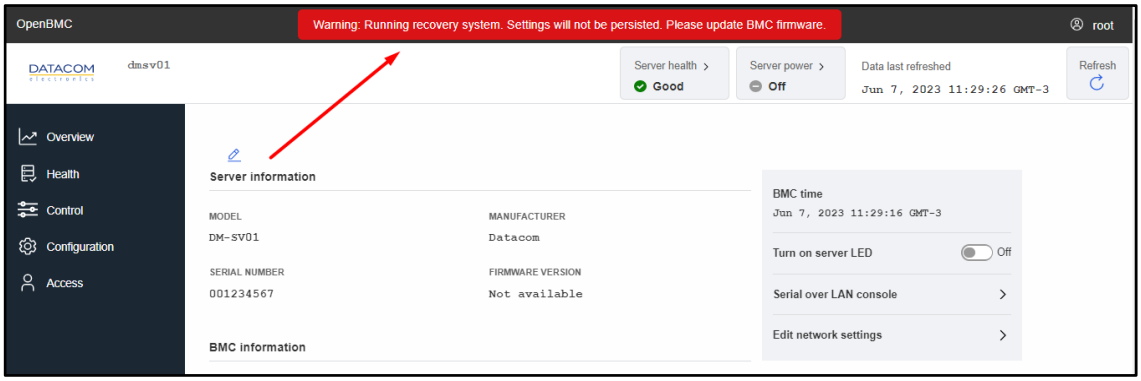


Figure 4: BMC running the recovery FW

6. Follow the procedure described in section “2.4.2.3 Factory Reset - BIOS and BMC” to perform a factory reset in the BMC.

   CAUTION: all the settings from the main BMC FW will be restored to factory defaults, including users and password configuration, network settings, etc.

7. Now perform a power cycle in the server, removing it from the DM1904 and inserting it again.
8. The main BMC FW will now initialize with factory defaults. The user can now access it with factory default network settings (dhcp) and login credentials (root/0penBMC), as explained in section “1.2 Getting started”.

1.4 Best practices for improving the BMC security

It’s highly recommended to follow the guidelines below as good practices for improving the security when accessing the BMC.

- Change the default password of the root user after the first login and use a strong new password. Create users for administering the BMC adequately according to the privilege level defined by your company’s policies. Details regarding the configuration of users and their respective privilege levels are described in section “2.5.2 Local users”.
- Avoid direct exposure of the BMC management port to the internet.
- Use separate networks for BMC management and for host LAN. If possible, use a dedicated network for BMC management with a restricted pool of IP addresses reserved for this purpose.
- It is recommended to use only the out-of-band dedicated Ethernet port for BMC management. If the in-band NC-SI interface needs to be used instead, it is advised to isolate the management network by using a dedicated VLAN for BMC management.
- Block outgoing BMC traffic to the internet using a firewall.
- Install a proper certificate for HTTP access to the BMC.
- Contact Datacom support team periodically to check for available BMC FW upgrades.

1.5 Redfish

The redfish is a RESTful (Representational State Transfer) application programming interface (API) developed by the DMTF (Distributed Management Task Force) and used for a broad range of converged infrastructure (CI) equipment, such as servers, storage devices, network equipment, etc.

The redfish was developed to improve the interoperability inside the data center facilities by providing a centralized and widely compatible way of managing different hardware resources by means of modern and scalable standards, specifications and interfaces.

The redfish API works over HTTP (Hypertext Transfer Protocol) and uses OData v4, which is the application layer protocol that defines the standards for the communication of the REST services by means of the HTTP. The redfish is also based on JSON (JavaScript Object Notation), which is a scalable data format that provides to customers a pattern that is easy to integrate to their management solution. The JSON schema is both human-readable and machine-compatible, making it easier to interpret and also to integrate to the IT management infrastructure.

The Refish interface is available in the DM-SV01 server as a remote management tool for accessing and controlling several resources available at the BMC. In the DM-SV01, the redfish API can be used for accessing and controlling a variety of resources of the BMC, such as inventory information, sensors data, host power actions, boot override configuration, users management, logs, FW update, etc. The most important actions available at the DM-SV01 server by means of the redfish interface are described in detail in section “3 Redfish API”.

The SNIA (Storage Networking Industry Association) provides an overview of Redfish on their website. Please refer to the link from reference (4) for having access to the content of their presentation. Additionally, please refer to the DMTF official website (5) for further information about the Redfish specification and additional documentation.

2 BMC Web GUI

2.1 Overview Menu

The “Overview” menu is used to show basic information about the system, as well as to provide some key shortcuts to the user. The “Figure 5” below shows an example of the Overview screen.

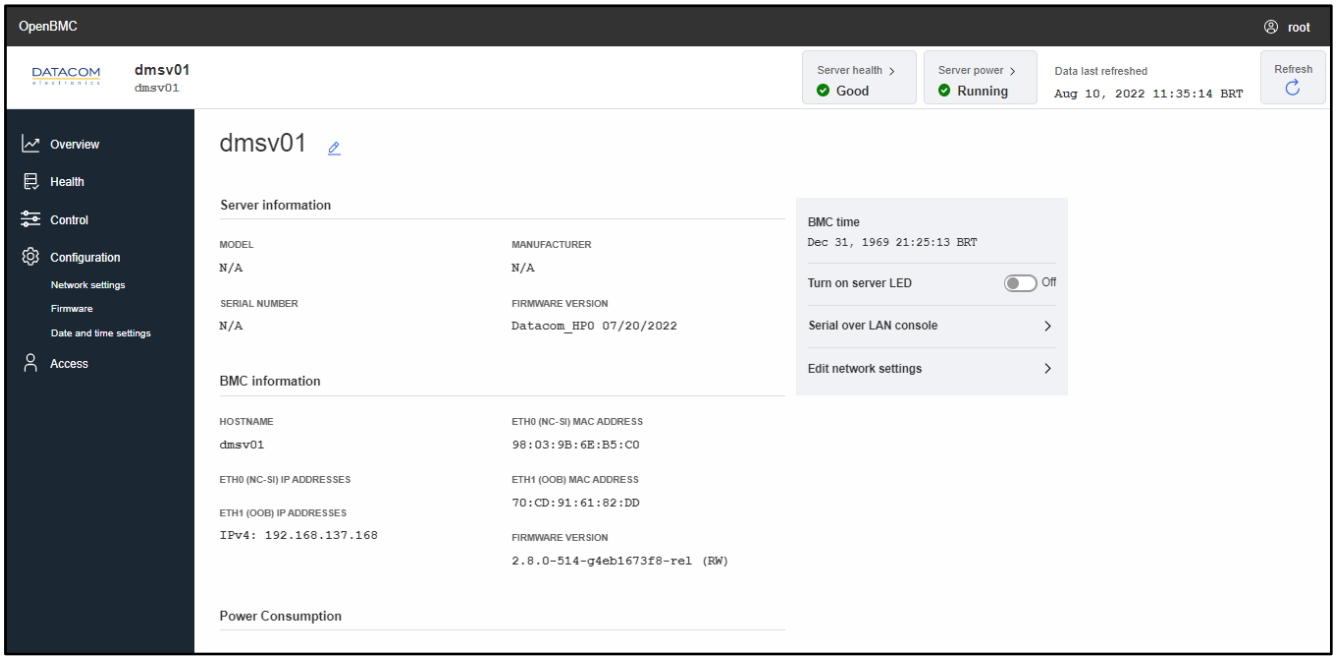


Figure 5: BMC Overview menu

The following information is shown at the “Overview” menu:

1. **Hostname.**
2. **Server information:** shows general information about the server, such as:
    a. Model.
    b. Manufacturer.
    c. Serial number.
    d. BIOS/UEFI firmware version.
3. **BMC information:** shows general information about the BMC, such as:
    a. Hostname.
    b. Mac addresses of the network interfaces eth0 and eth1.
    c. IP addresses of the network interfaces eth0 and eth1.
    d. BMC firmware version.
4. **Power Consumption:** displays the power consumption of the server (the value shown corresponds to the power consumption of the server when the overview page was last loaded).
5. **High priority events:** displays high priority events if available.

The shortcuts below are available in the overview menu:

1. **Turn on server LED:** this option turns the server LED on or off. For detailed information regarding the functionality and use cases of this LED, please refer to the section “2.3.2 Server LED”.
2. **Serial over LAN console:** redirects the user to the “Serial over LAN console” menu. Details regarding this function can be found in section “2.3.4 Serial over LAN console”.

3. **Edit network settings:** redirects the user to the “Network settings” menu. Details regarding this function can be found in section “2.4.1 Network settings”.

Additionally, there are two buttons in the header of the web page:

- **Server Health:** provides information about the health state of the server, indicating if there is some critical alarm or not. Details of this function can be found in section “2.2.1.1 Event Logs through Server health button”. The state of the server health button is updated only after the web page is refreshed.
- **Server Power:** provides information about the power state of the server, indicating if it is powered on and running or powered off. When clicking on this button, the user is redirected to the Server Power Operations menu. The functions present in this menu are detailed in section “2.3.1 Server power operations”.

2.2 Health Menu

2.2.1 Event log

The events or alarms triggered in the system are identified by the BMC and logged in the database, allowing the user to check them later by means of the “Event log” menu. An example of an event log from the DM-SV01 server can be seen in Figure 6 and Figure 7.

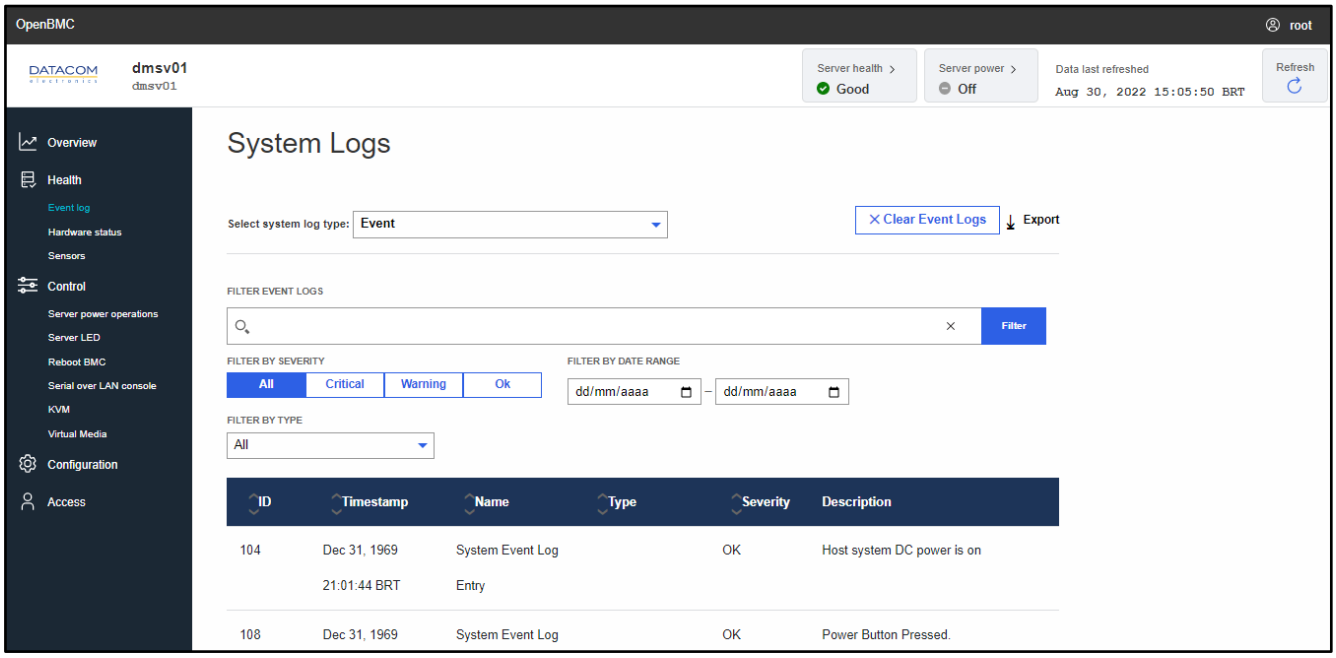


Figure 6: BMC Event log menu

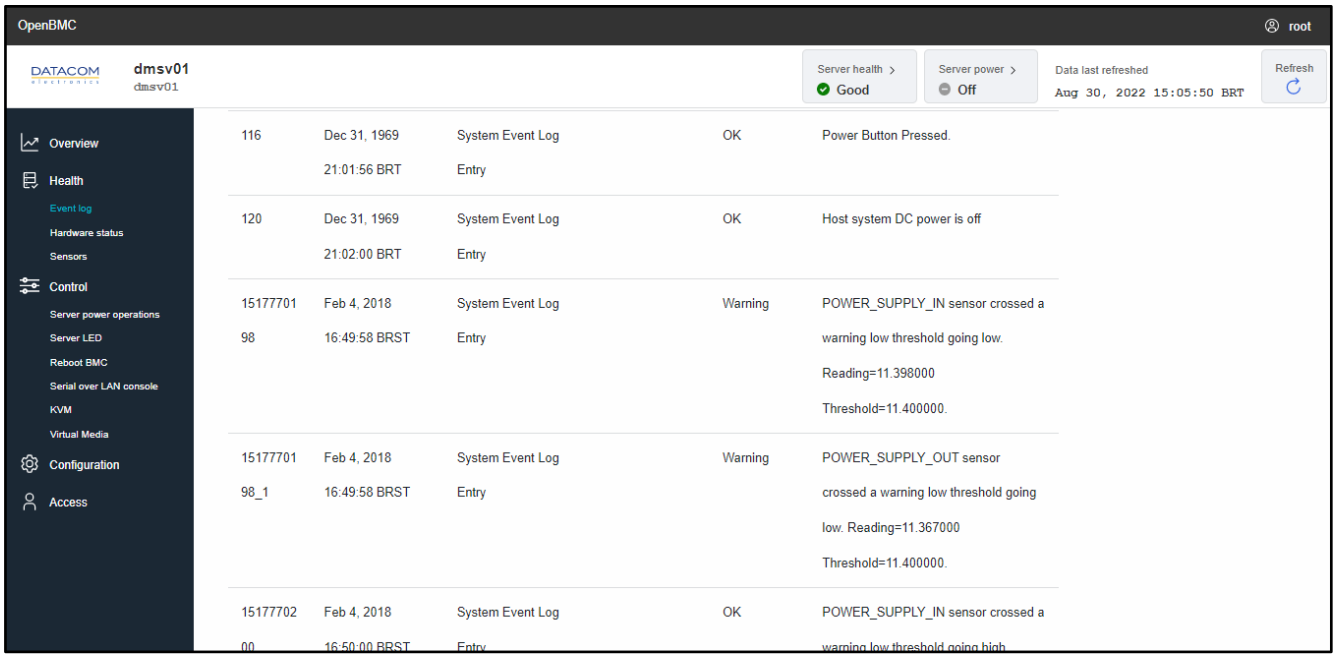


Figure 7: BMC Event log menu list

In order to help the user to find an event of interest, the following filtering options are available:

1. System log type:
    a. SEL (System Event Log)
    b. Event
    c. Oem
2. Filter by severity:
    a. All
    b. Critical
    c. Warning
    d. OK
3. Filter by date.
4. Filter by type, if applicable.

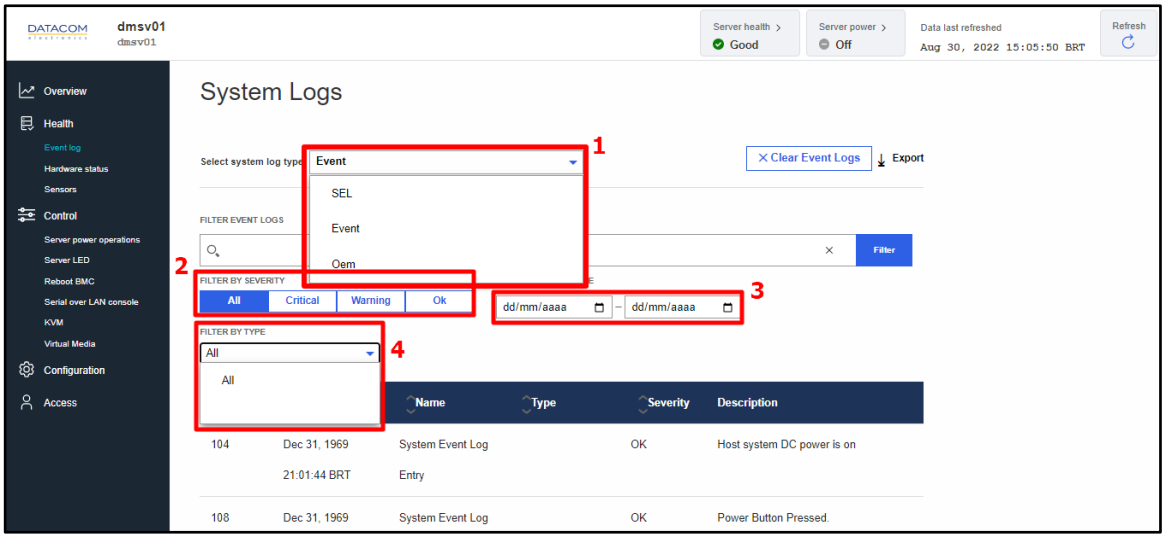


Figure 8: BMC Event log filters

2.2.1.1 Event Logs through Server health button

The event logs from type “Event” can also be accessed by means of the “Server Health” button present in the header of the OpenBMC web page, as shown in the Figure 9.

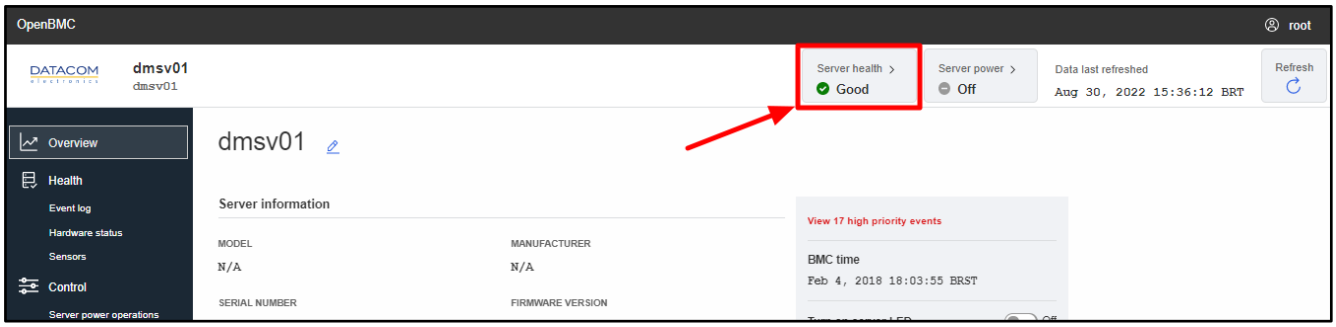


Figure 9: Server health button

The web page provides the same filters as shown in the standard “Event log” screen, but the events are displayed in a different way, as shown in the Figure 10.

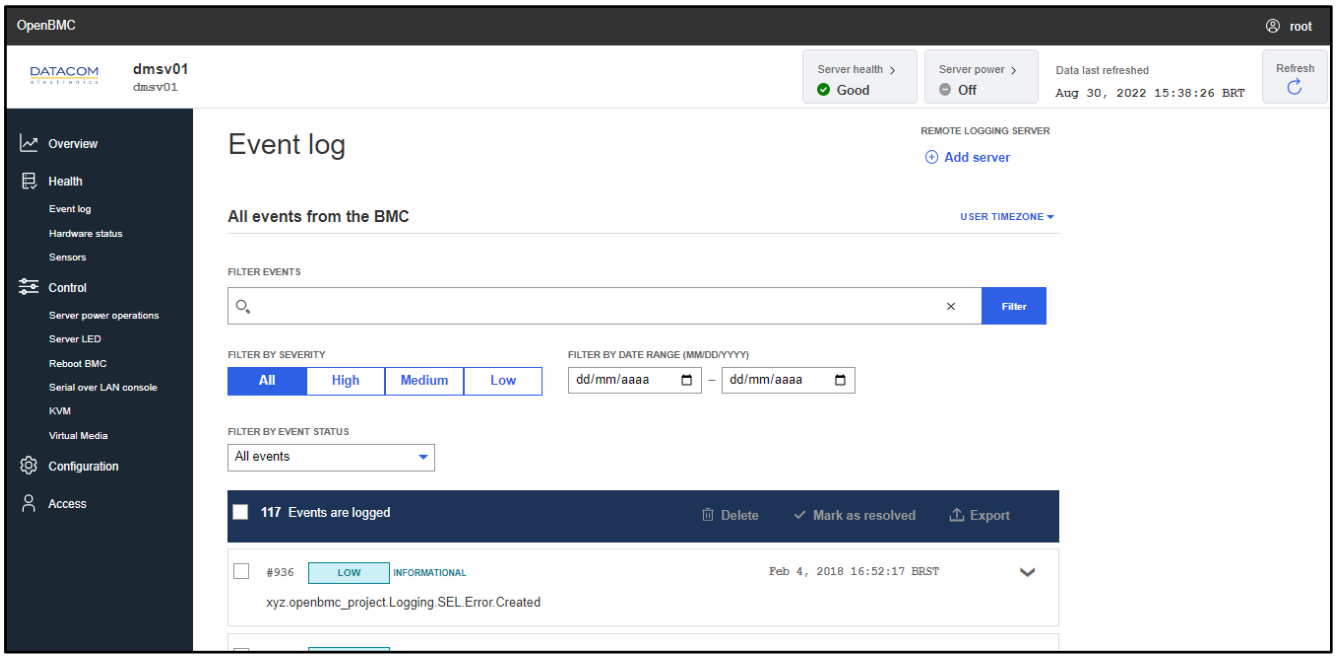


Figure 10: Server health button web page

The event log viewer from this web page provides the following action possibilities for the user (please refer to Figure 11):

1. The user may select one or more events from the event list by clicking in the individual mark boxes or select all the events at once by clicking in the mark box present on the header of the list.
2. The user can delete an event from the log, so that it will be completely removed from the database.
3. The user can mark an event as resolved. When doing this, the event will still be available in the database, but it will not cause any alarm indication in the system.

4. The user can export the event log. When this option is used, the BMC creates a ".json" file containing the events selected and provides the download of the file.

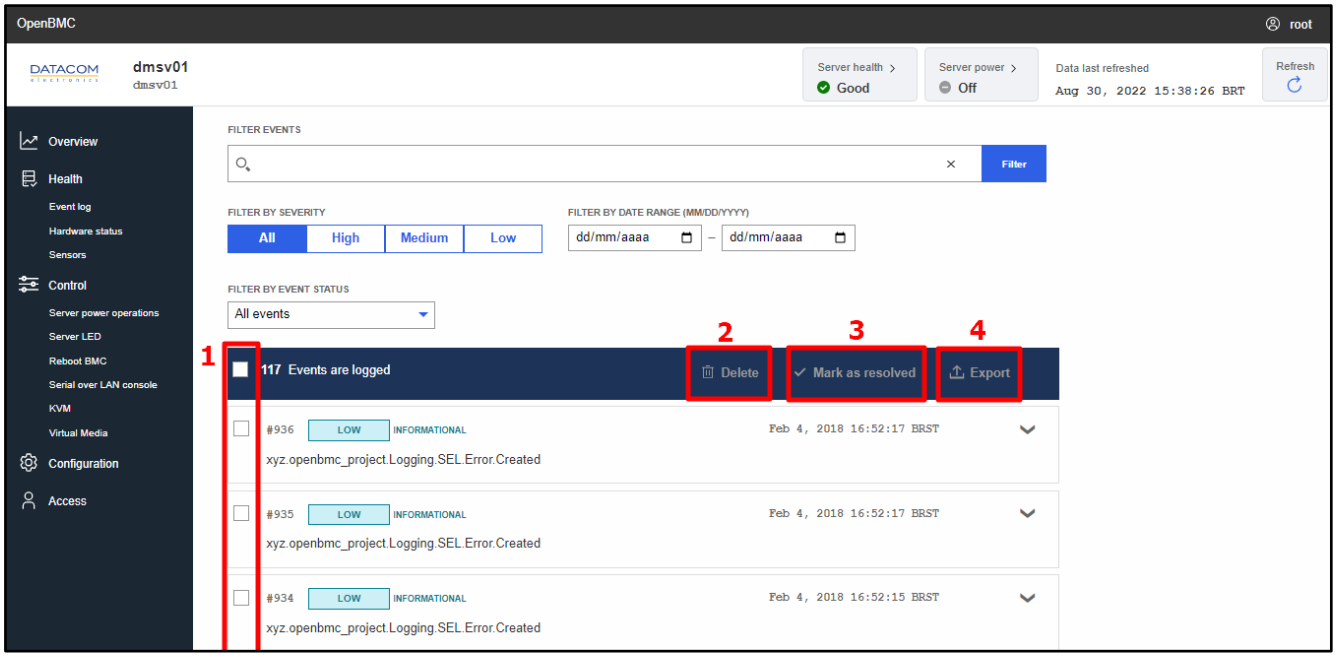


Figure 11: Server health button web page options

2.2.1.2 Remote logging server

When accessing the event logs from the BMC by means of the server health button, the "Remote Logging Server" option is available, as shown in Figure 12. This option allows the user to configure a remote server to receive the logs from the BMC.

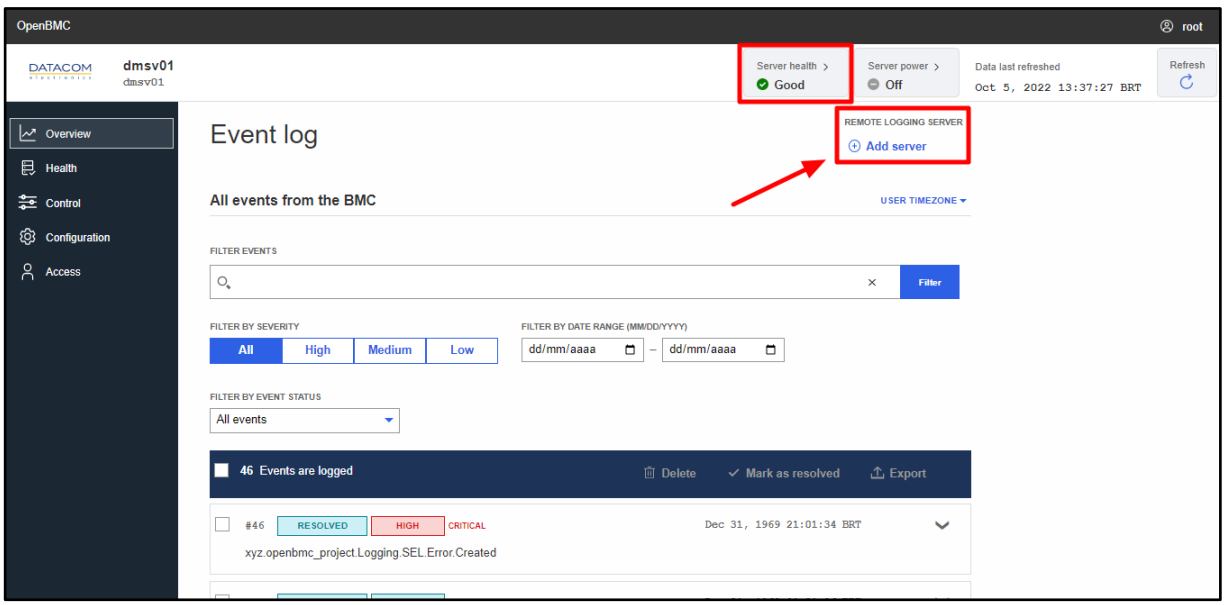


Figure 12: Remote Logging Server button

When clicking in the “Add server” button, the user is prompted to configure the IP address and port number of the remote logging server, as shown in Figure 13.

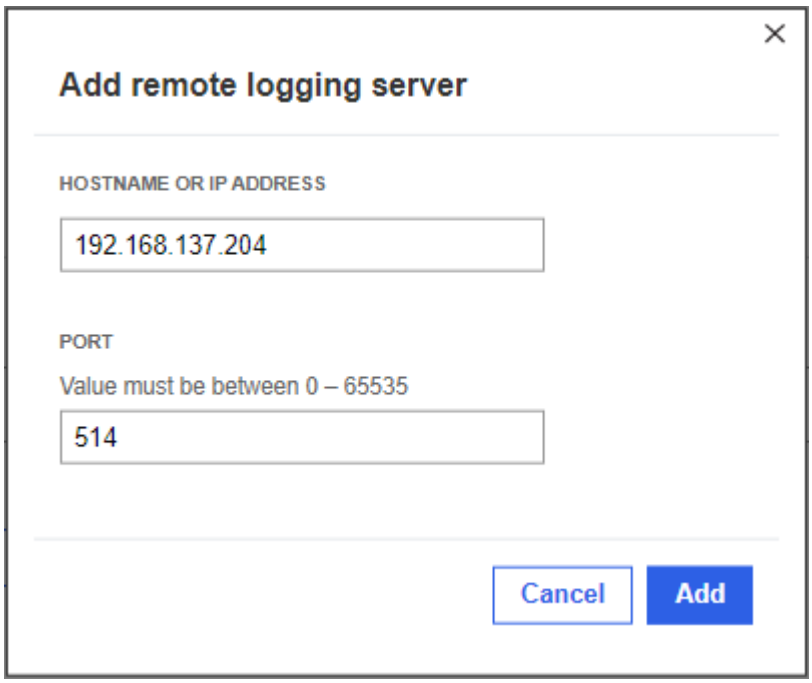


Figure 13: Remote Logging Server configuration

Once the remote server settings are configured, the BMC starts to send its logs to the destination IP address and port. The remote server must be properly configured to accept the log packets sent by the BMC. As an example, a “rsyslog” server can be installed in an Ubuntu OS and configured to collect and save the BMC logs (https://www.rsyslog.com/).

2.2.2 Hardware status

The “Hardware status” menu is used to check the inventory information of the following system components:

- Mainboard
- NVMes
- DIMMs
- Mezzanine card
- CPUs
- Riser Cards
- 2xE1.S Adapter Card

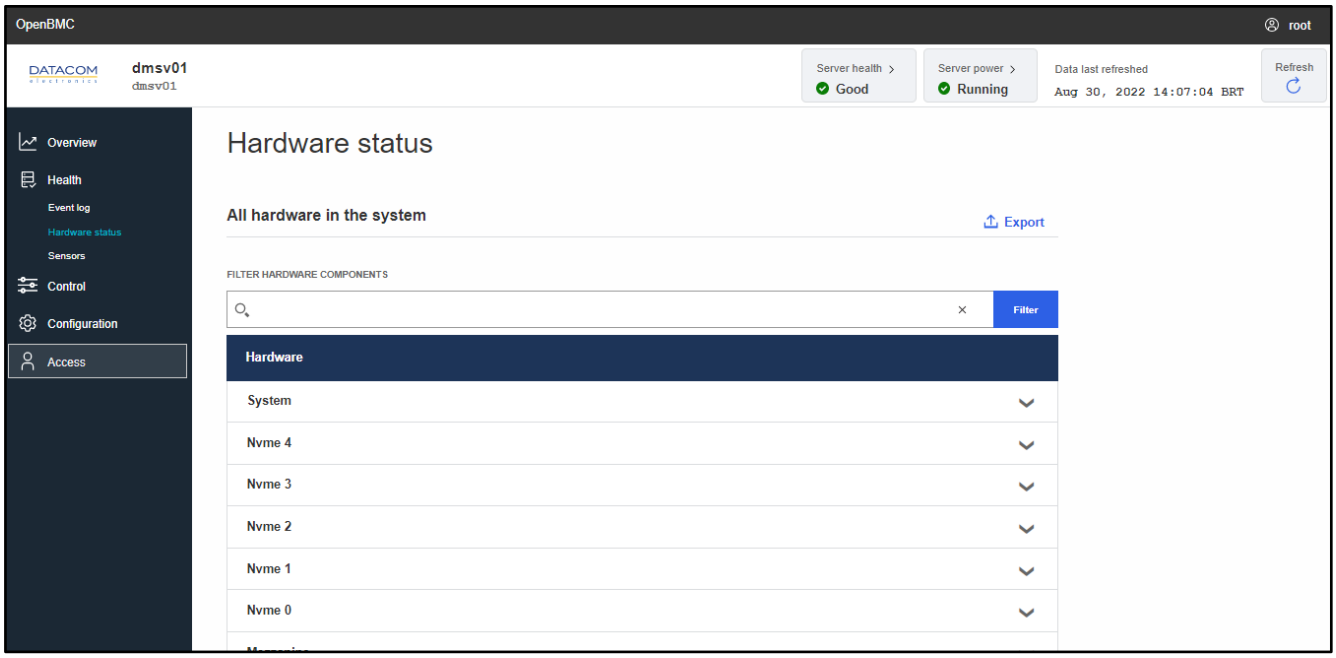


Figure 14: Hardware status menu

The data available for each component depends on the component type. In general, the user can access information such as presence, model, part number, serial number, etc. The figure below shows an example of the inventory data for one of the CPUs compatible with the DM-SV01 server.

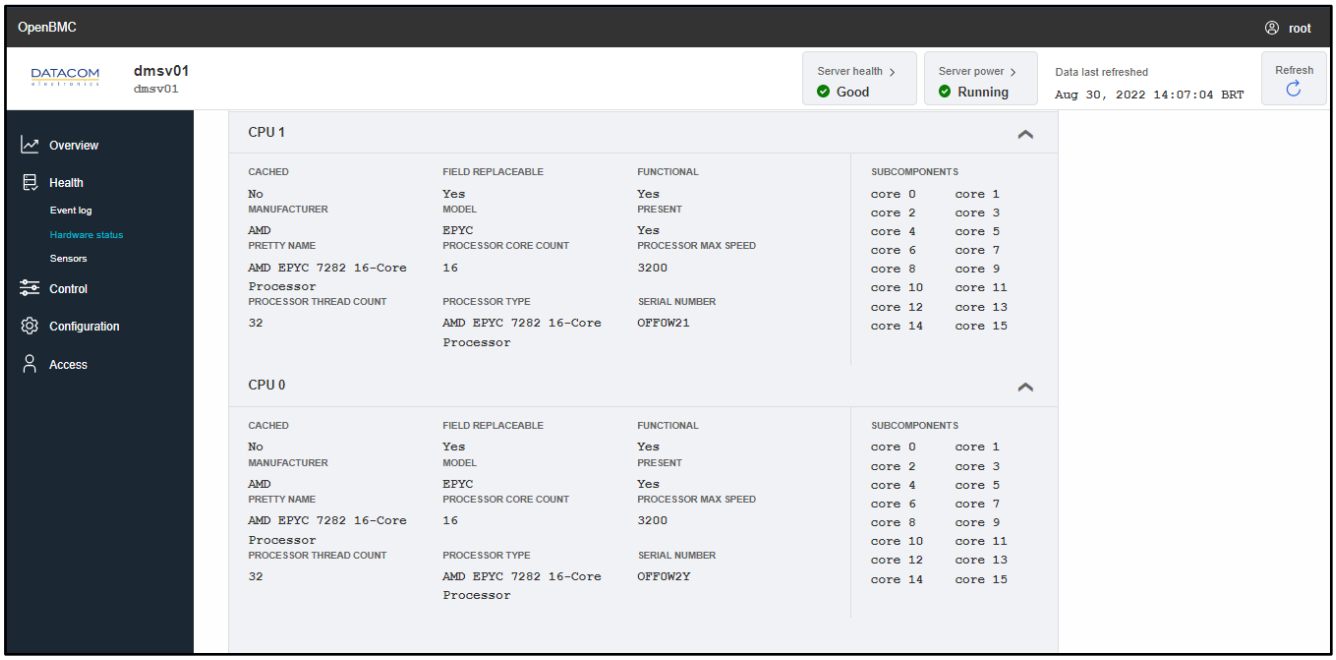


Figure 15: Hardware status menu - CPUs view

The NVMe devices are listed in the menu by following a predefined numbered sequence. The NVMe device “nvme0” is the onboard M.2 SSD. The NVMe devices “nvme1” up to “nvme10” are the E1.S SSDs, which are arranged in the server as shown in the Figure 16, when using the Riser Card 3x8.

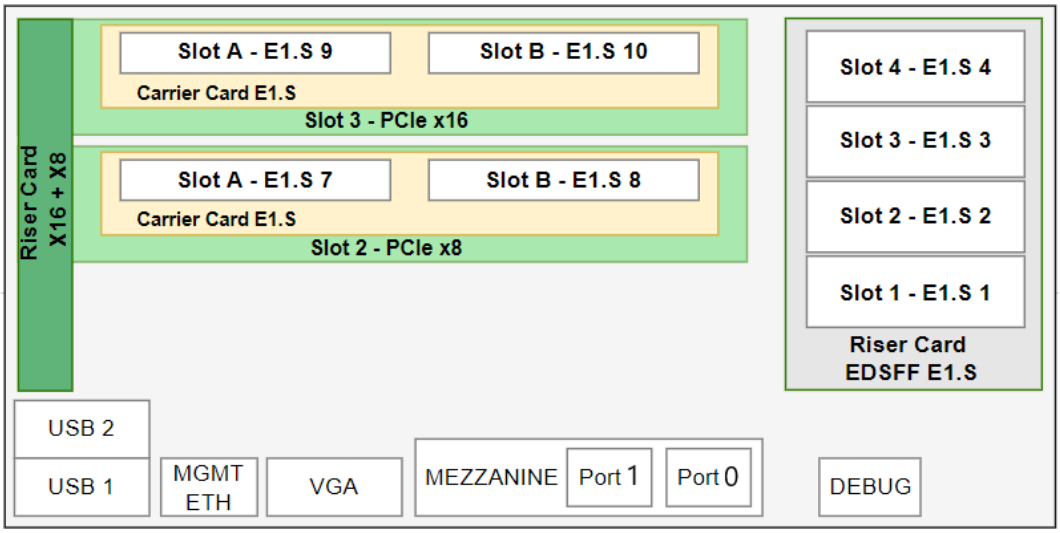


Figure 16: Risercard PCIe 3x8 equipped with 2x E1.S PCIe Adapter Card

If the Riser Card X16+X8 is used, on the other hand, the numbering of the E1.S NVMe devices follows the order depicted in the Figure 17.

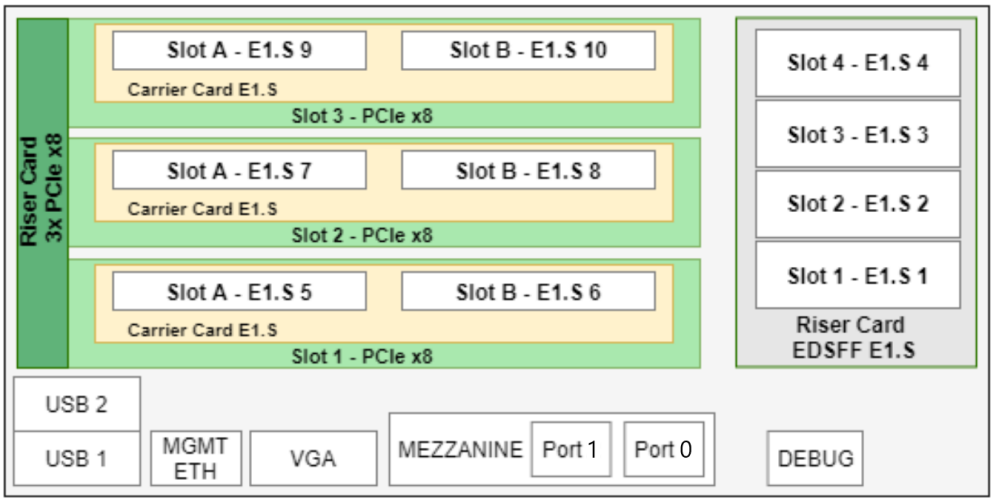


Figure 17: Risercard PCIe x16+x8 equipped with 2x E1.S PCIe Adapter Card

The DDR memories also follow a predefined numbering sequence, as shown in the table below.

DIMM name	DIMM slot in the server
DIMM 0	Information about DDR memory installed at socket P0, slot A
DIMM 1	Information about DDR memory installed at socket P0, slot B
DIMM 2	Information about DDR memory installed at socket P0, slot C
DIMM 3	Information about DDR memory installed at socket P0, slot D
DIMM 4	Information about DDR memory installed at socket P0, slot E

DIMM 5	Information about DDR memory installed at socket P0, slot F
DIMM 6	Information about DDR memory installed at socket P0, slot G
DIMM 7	Information about DDR memory installed at socket P0, slot H
DIMM 8	Information about DDR memory installed at socket P1, slot A
DIMM 9	Information about DDR memory installed at socket P1, slot B
DIMM 10	Information about DDR memory installed at socket P1, slot C
DIMM 11	Information about DDR memory installed at socket P1, slot D
DIMM 12	Information about DDR memory installed at socket P1, slot E
DIMM 13	Information about DDR memory installed at socket P1, slot F
DIMM 14	Information about DDR memory installed at socket P1, slot G
DIMM 15	Information about DDR memory installed at socket P1, slot H

Table 1: DIMMs mapping on Hardware status menu

Important: some system components are detected by BIOS/UEFI only during the system boot process, when the information is then sent to the BMC through IPMI. Therefore, some system components may not appear in the “Hardware status” menu or may be outdated if the system was not yet initialized. Please perform a power on in the host CPUs (please check the section “2.3.1 Server power operations”) and wait for the boot process to complete to make sure the information available in this menu is properly updated.

2.2.3 Sensors

The DM-SV01 server has a set of sensors which are responsible for monitoring the voltage, current and temperature at several relevant spots of the system. The sensors information is used for controlling the FANs speed and generating alarms or emergency shutdown in case of reaching a critical threshold.

The Figure 18 below shows the “Sensors” screen available at the BMC web management interface.

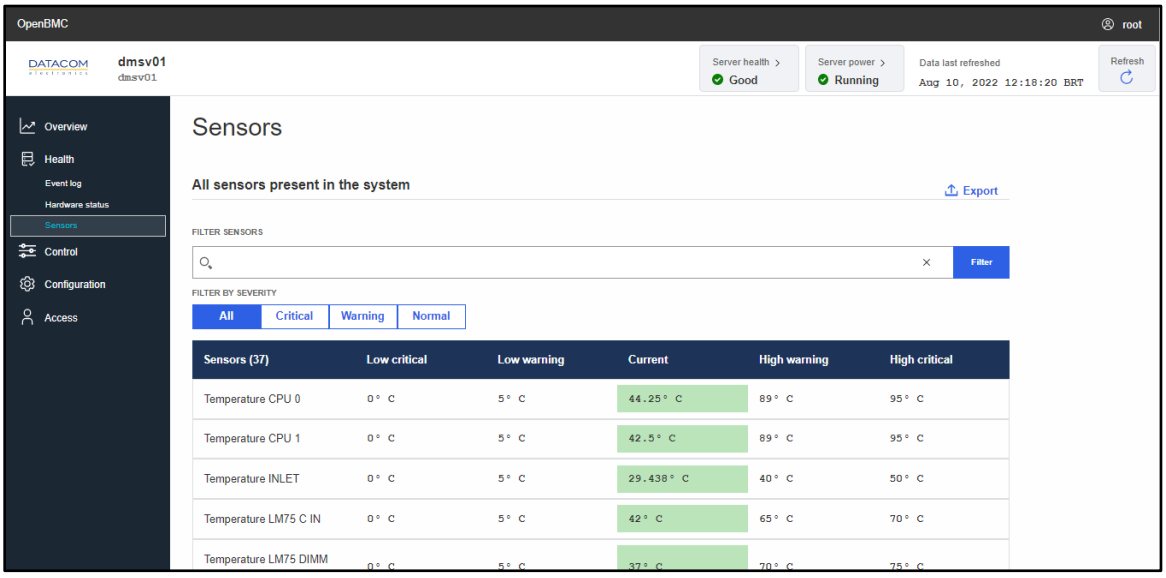


Figure 18: BMC Sensors menu

The figure below shows the location of the temperature sensors in the DM-SV01 mainboard.

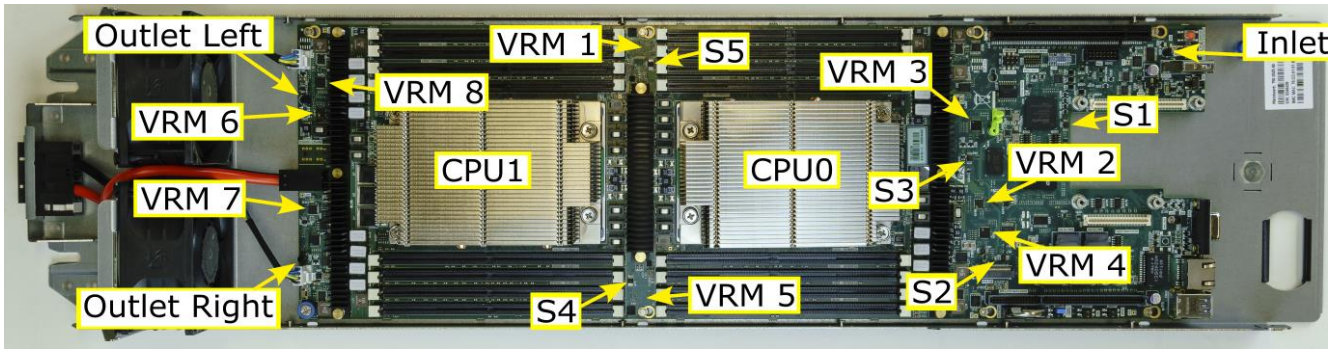


Figure 22: DM-SV01 temperature sensors location

The table below shows the description of each sensor and its respective alias in the BMC web management interface.

Sensor name	Sensor name in the BMC web GUI	Sensor positioning
CPU 0	Temperature CPU 0	Processor P0
CPU 1	Temperature CPU 1	Processor P1
Inlet	Temperature INLET	Airflow Input
Outlet Left	Temperature OUTLET LEFT	Airflow Output - Left
Outlet Right	Temperature OUTLET RIGHT	Airflow Output - Right
VRM1	Temperature VRM P0 VDD CORE RUN	Voltage Regulator Module 1 - CPU P0 Core Power
VRM2	Temperature VRM P0 VDD SOC RUN	Voltage Regulator Module 1 - CPU P0 SoC Power
VRM3	Temperature VRM P0 VDD MEM ABCD SUS	Voltage Regulator Module 1 - CPU P0 ABCD DIMMs Power
VRM4	Temperature VRM P0 VDD MEM EFGH SUS	Voltage Regulator Module 1 - CPU P0 EFGH DIMMs Power
VRM5	Temperature VRM P1 VDD CORE RUN	Voltage Regulator Module 1 - CPU P1 Core Power
VRM6	Temperature VRM P1 VDD SOC RUN	Voltage Regulator Module 1 - CPU P1 SoC Power
VRM7	Temperature VRM P1 VDD MEM ABCD SUS	Voltage Regulator Module 1 - CPU P1 ABCD DIMMs Power
VRM8	Temperature VRM P1 VDD MEM EFGH SUS	Voltage Regulator Module 1 - CPU P1 EFGH DIMMs Power

S1	Temperature LM75 MEZZ	Sensor 1 - near the mezzanine card
S2	Temperature LM75 LEFT IN	Sensor 2 - near the airflow input on left side
S3	Temperature LM75 C IN	Sensor 3 - near the airflow input on center
S4	Temperature LM75 DIMM ML	Sensor 4 - near DIMMs on left side
S5	Temperature LM75 DIMM MR	Sensor 5 - near DIMMs on right side
-	Temperature Nvme[x]	Temperature of the nvme module. Each nvme module is monitored and the temperature is shown in the BMC web GUI as nvme0, nvme1, etc.
-	Temperature MEZZANINE CARD	Temperature of the mezzanine card, when connected.
-	Temperature PCIE CARD SLOT3	Temperature of the PCIe card from riser card slot 3, when connected.

Table 2: DM-SV01 temperature sensors description

Additionally, the DM-SV01 is also capable of monitoring the system input power, the FANs speed and some primary voltages. The table below shows the additional monitoring outputs of the DM-SV01.

Sensor name in the BMC web GUI	**Sensor description**
Fan Tach FAN LEFT	Left fan speed
Fan Tach FAN RIGHT	Right fan speed
Voltage POWER SUPPLY IN	12V Power supply input (before the input power controller)
Voltage POWER SUPPLY OUT	12V Power supply output (after the intput power controller)
Voltage VDD 5 DUAL	5V Power supply
Voltage VDD 33 DUAL	3.3V Standby Power Supply
Voltage VDD 33 RUN	3.3V Active Power Supply
Current POWER SUPPLY	DM-SV01 server total input current
Energy TOTAL ENERGY	Total energy consumed by the system, in “joules”
Power PEAK POWER	Maximum instantaneous power value consumed by the system
Power Total Power	DM-SV01 server total input power

Table 3: DM-SV01 additional sensors description

2.2.3.1 Power consumption sensors

The DM-SV01 server has some sensors that are capable of monitoring the power consumption of the system. The following power sensors are available:

- **Total Energy:** this sensor displays the accumulated energy consumed by the DM-SV01 server in “joules”. The measurement is cumulative and it comprises the whole energy consumed by the system since the server has been turned on until the current moment. The user can reset the total energy counter at any time, in order to monitor the energy consumption in the desired time period. For details regarding the reset of the sensor, please refer to section “2.2.3.1.1 Power sensors Reset”.
- **Peak Power:** this sensor measures the maximum instantaneous power consumed by the server since it has been turned on. The value of this sensor will be updated only if the BMC measures a peak power value higher than the current value being shown in the sensor. The peak power sensor can also be reset by the user whenever necessary. For details regarding the reset of the sensor, please refer to section “2.2.3.1.1 Power sensors Reset”.
- **Total Power:** this sensor shows the instantaneous power being consumed by the server. In the BMC web GUI, the value of the sensor is updated only when the web page is refreshed.

Important: if the “Total Energy” and “Peak Power” sensors are not available in the BMC, please update the BMC SW by following the procedure described in section 2.4.2.2 FW update process - BMC or BIOS.

Energy TOTAL ENERGY	0 JOULES	0 JOULES	6183689.611526 JOULES	9223372036854.775 JOULES	9223372036854.775 JOULES
Power PEAK POWER	0 WATTS	0 WATTS	255.691854 WATTS	732 WATTS	780 WATTS
Power Total Power	0 WATTS	0 WATTS	86.162904 WATTS	732 WATTS	780 WATTS

Figure 19: BMC power sensors

2.2.3.1.1 Power sensors Reset

The “Total Energy” and “Peak Power” sensors can be reset whenever necessary, allowing the user to measure the power consumption of the server during a controlled period of time. Using the reset, the user can control the starting point of the power monitoring and then use the BMC web GUI to read the accumulated power consumed by the system from the reset moment until the current time.

The reset function for the power sensors is available at the “Firmware” menu of the BMC, as shown in Figure 20.

Important: if the “Total Energy” and “Peak Power” sensors are not available in the BMC, please update the BMC SW by following the procedure described in section 2.4.2.2 FW update process - BMC or BIOS.

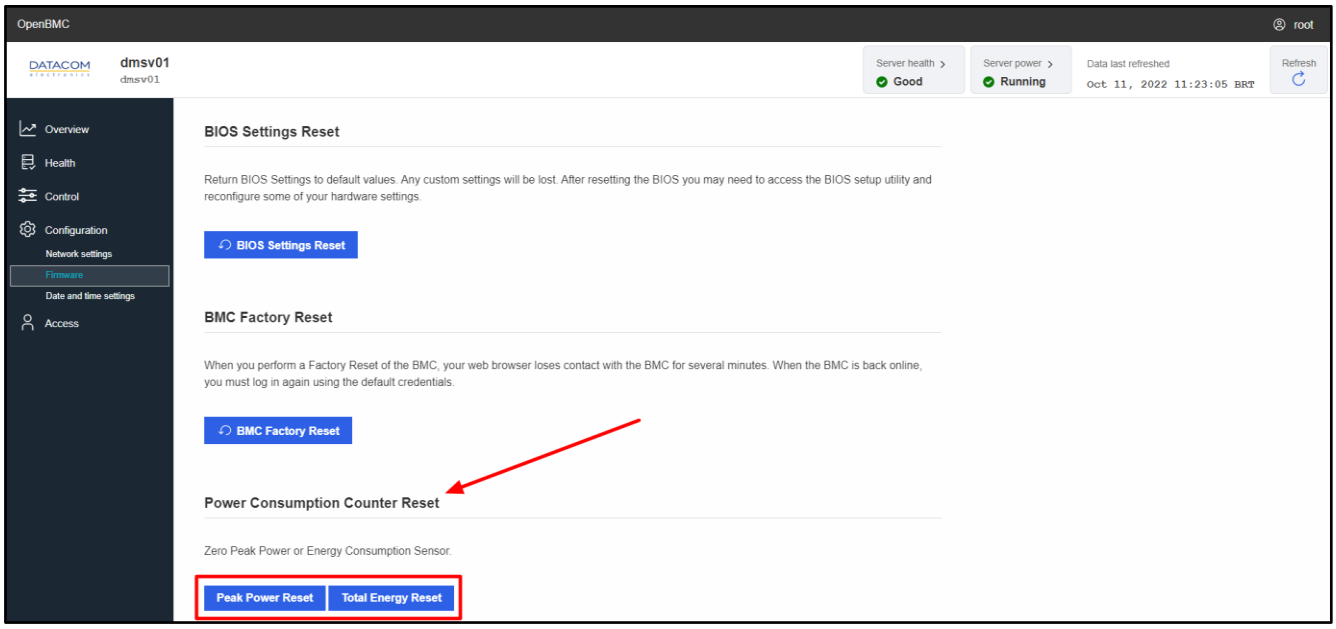


Figure 20: Reset function for the power sensors

2.3 Control Menu

2.3.1 Server power operations

2.3.1.1 Operations

The Operations menu is used to turn the host CPUs on or off and also to perform a reboot. The following options are available:

- **Power on:** power on the host processors.
- **Orderly Reboot:** causes the OS controlled reboot process to be triggered.
- **Immediate Reboot:** causes the CPU to be abruptly rebooted, without previous warning to the OS.
- **Orderly Shutdown:** causes the OS controlled shutdown process to be triggered. After the shutdown procedure has been completed, the server will switch to the STANDBY Mode.
- **Immediate Shutdown:** causes the CPU to be abruptly shut down, without previous warning to the OS. After the shutdown procedure has been completed, the server will switch to the STANDBY Mode.

The shut down and reboot processes do not interfere with the BMC operations. The BMC operation is independent of the host CPUs power, so the BMC keeps working even if the host CPUs are powered off or rebooted.

The procedure to perform a reboot or shutdown is pretty straightforward. The user selects between the options “orderly” or “immediate” by marking the corresponding checkbox and then click on the respective “Reboot” or “Shut down” button.

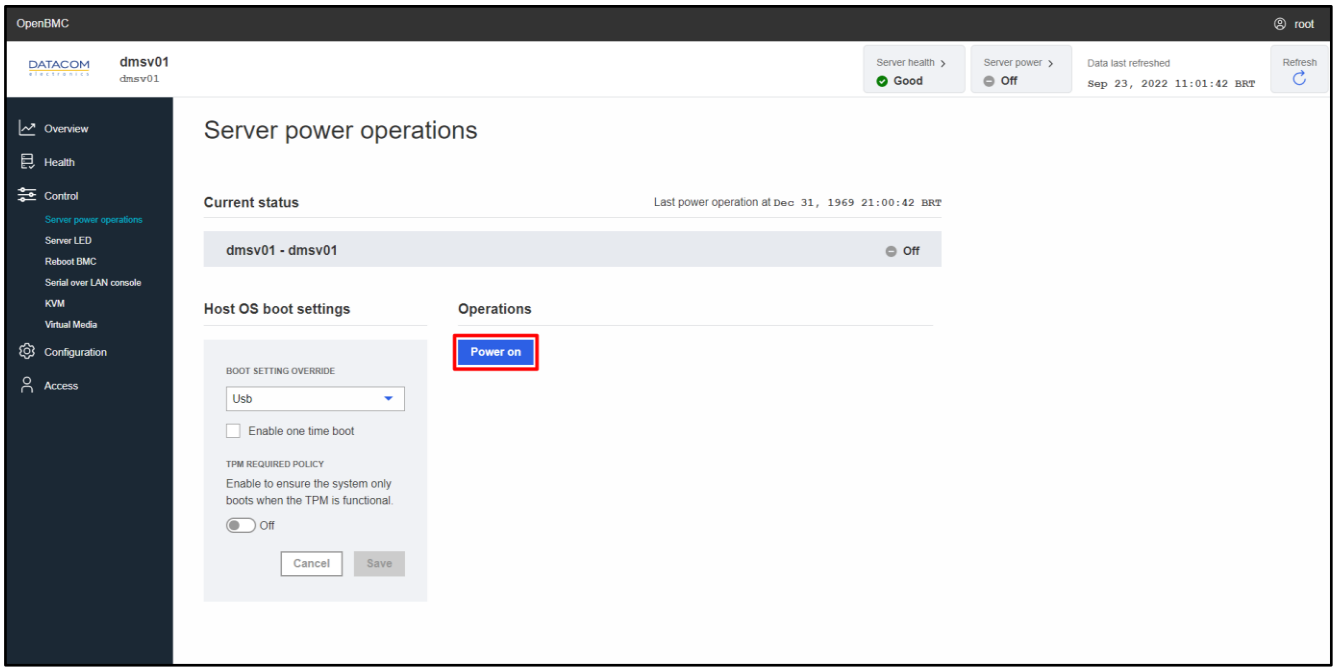


Figure 21: Server Power Operations menu when the host processors are powered off

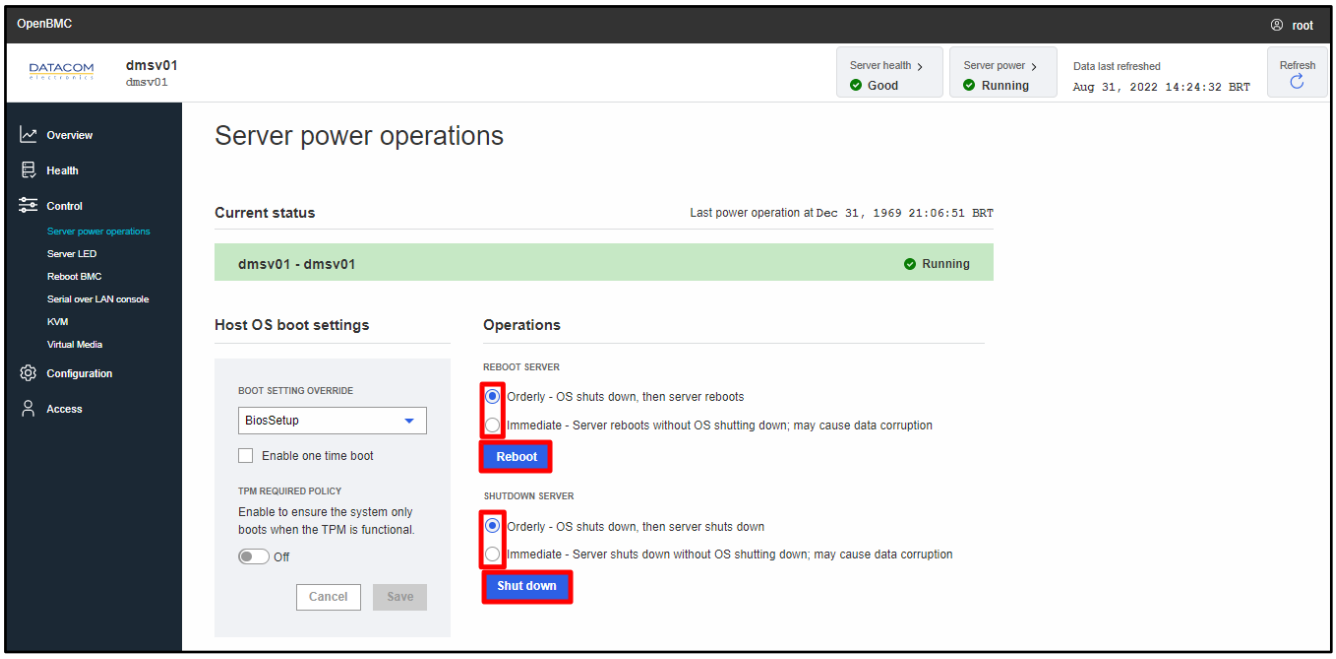


Figure 22: Server Power Operations menu when the host processors are powered on

2.3.1.2 Host OS boot settings - Boot override

The “Host OS boot settings” is used to configure the boot override option, which is used to select the main boot option for the system without the need to access the BIOS/UEFI menu.

Once a boot option other than “None” is selected in the “Boot Setting Override”, this setting will override the boot sequence configured in the UEFI/BIOS menu. However, when “None” is selected, the boot override is disabled and the boot sequence defined in the BIOS/UEFI menu is used. The following boot options are available in the BMC web GUI:

- **None:** no boot override configured. Boot priority is defined in the BIOS/UEFI menu.
- **Pxe:** pxe boot. Used for booting through the network (netboot).
- **Hdd:** Hard Disk Drive, when available.
- **Cd:** Cd or Virtual Media, when available.
- **Diags:** Diagnostic boot.
- **BiosSetup:** used for automatically entering the BIOS menu when booting.
- **Usb:** usb device, when available.

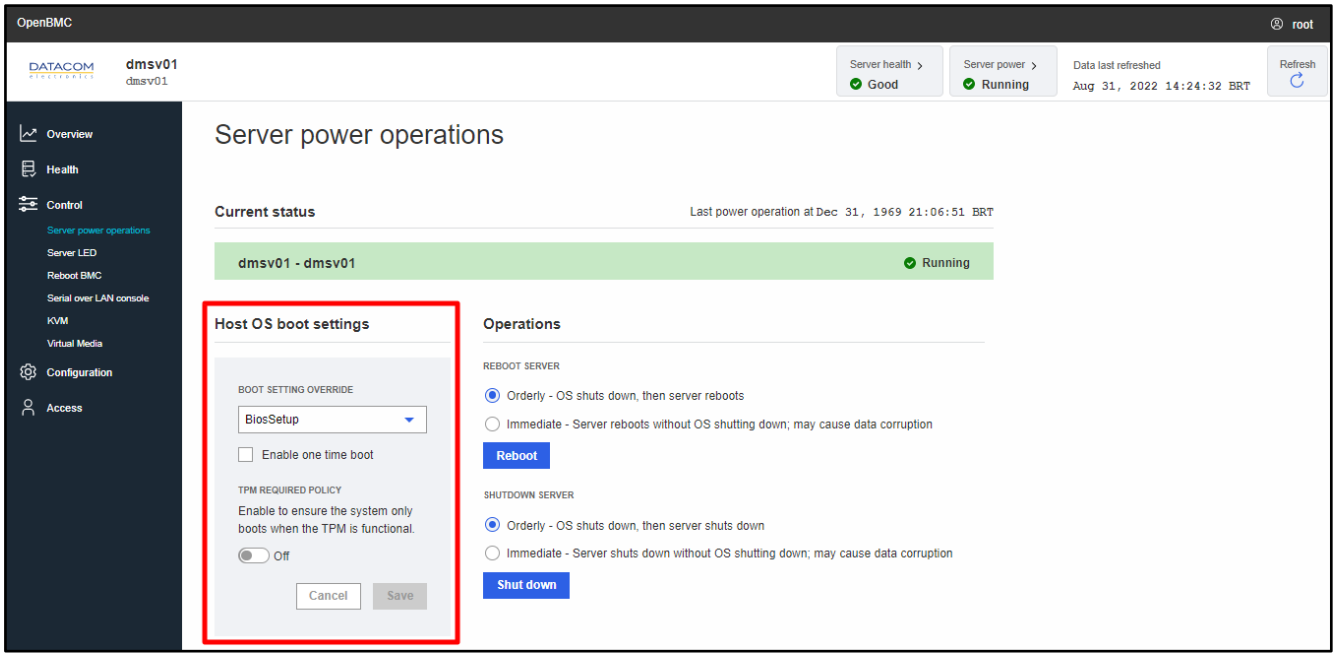


Figure 23: Host OS boot settings menu

There are two additional options available in the “Host OS boot settings” menu:

- The “Enable one time boot” checkbox, when active, configures only the next boot to use the override settings defined in the “Boot Setting Override” drop down list. As soon as the first boot is completed, the boot override option is disabled, returning automatically to “None”.
- The “TPM Required Policy” can be enabled in order to ensure that the system will boot only if the TPM (Trusted Platform Module) is active and functional. Please refer to the “DM-SV01 BIOS Manual” (2) for details regarding the TPM settings and refer to the “DM-SV01 Product Manual” (1) for additional information about the TPM functionality.

After configuring any of the options mentioned above, the user can apply the settings by clicking on the “Save” button or discard them by clicking on the “Cancel” button.

2.3.2 Server LED

The “ID LED”, or “Server LED” can be controlled via the ID button in the front panel of the DM-SV01 server and also via the BMC WEB management interface. For details regarding the physical ID button, please refer to the DM-SV01 Product Manual (1).

The Figure 24 below shows the virtual button used to turn the ID LED on or off in the BMC web management GUI.

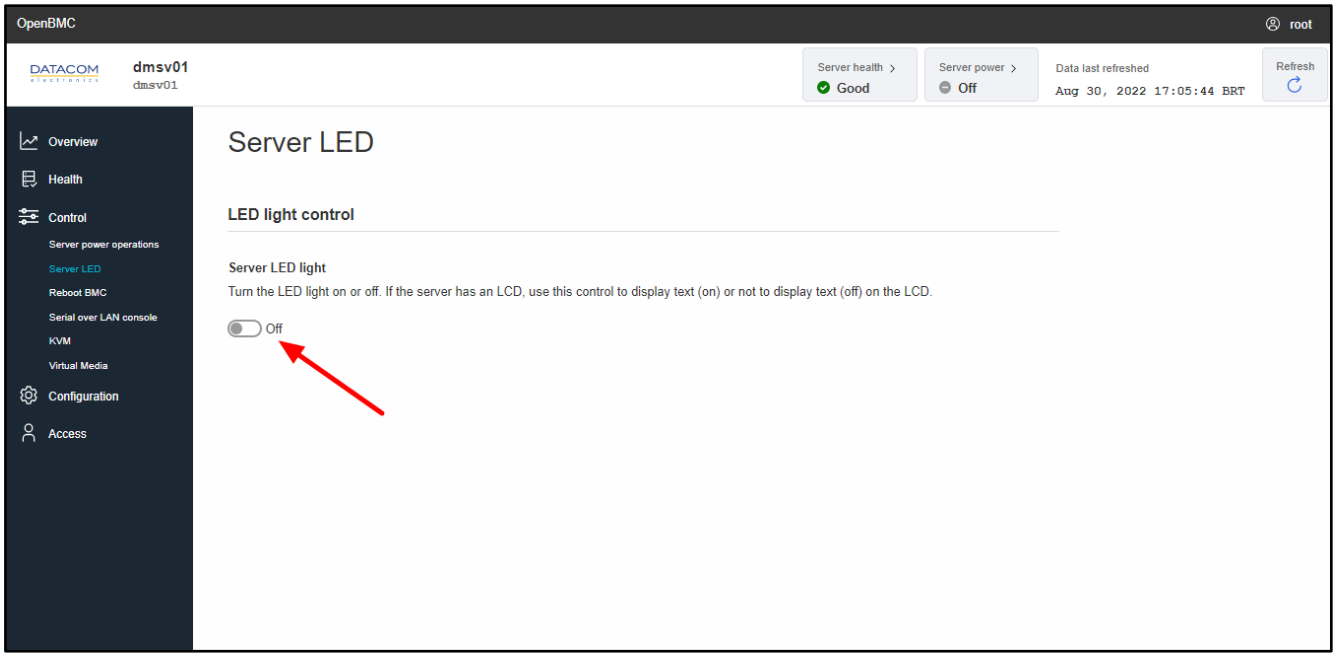


Figure 24: Server LED on/off button

When the ID LED is set to “on”, the LED shown in Figure 26 starts blinking and allows the datacenter staff to properly identify the server inside the facilities.

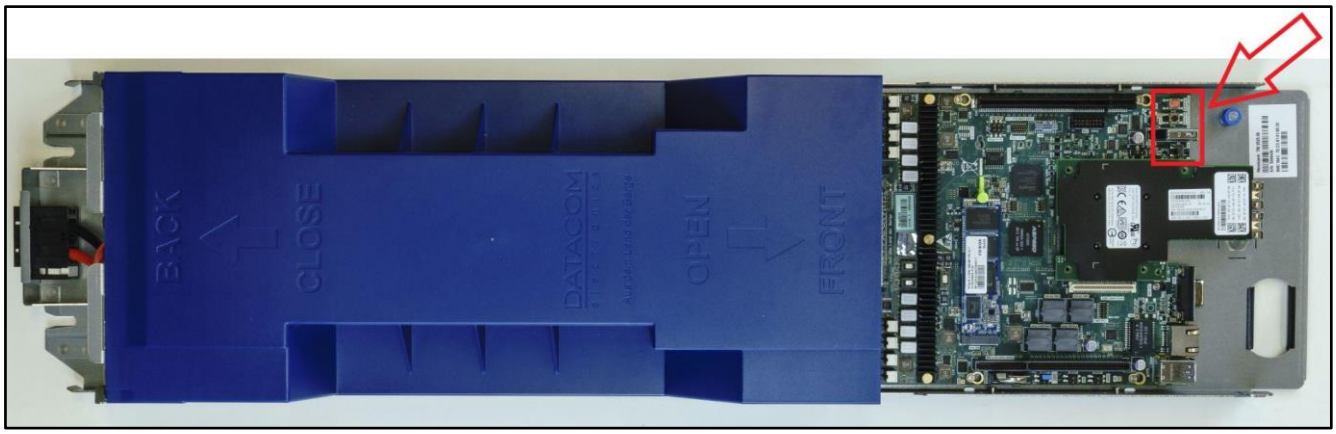


Figure 25: DM-SV01 LEDs location

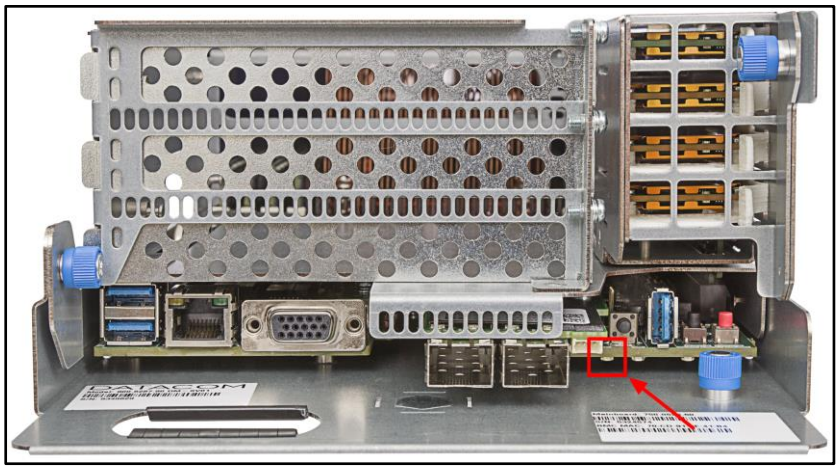


Figure 26: ID LED position in the DM-SV01 front panel

Important: the ID LED can also be physically activated by means of the ID button - details of this functionality can be found at the “DM-SV01 Product Manual” (1). When the ID LED is turned on or off by means of the ID button, the “Server LED light” button in the BMC web GUI will update its status only after the web page is refreshed.

2.3.3 Reboot BMC

The button shown in Figure 27 is used to reboot the BMC. Please note that the BMC and the host processors are independent of each other, so rebooting the BMC will not affect the CPUs, memories, disks and the workloads running on them. The only effect is that the user will lose access to the management functionalities from the BMC while it is in the reboot process.

The reboot process also does not reset any configuration from the BMC. All the settings (including network configuration) and event log information will be preserved and the BMC web management interface will be available again as soon as the BMC completes the booting phase.

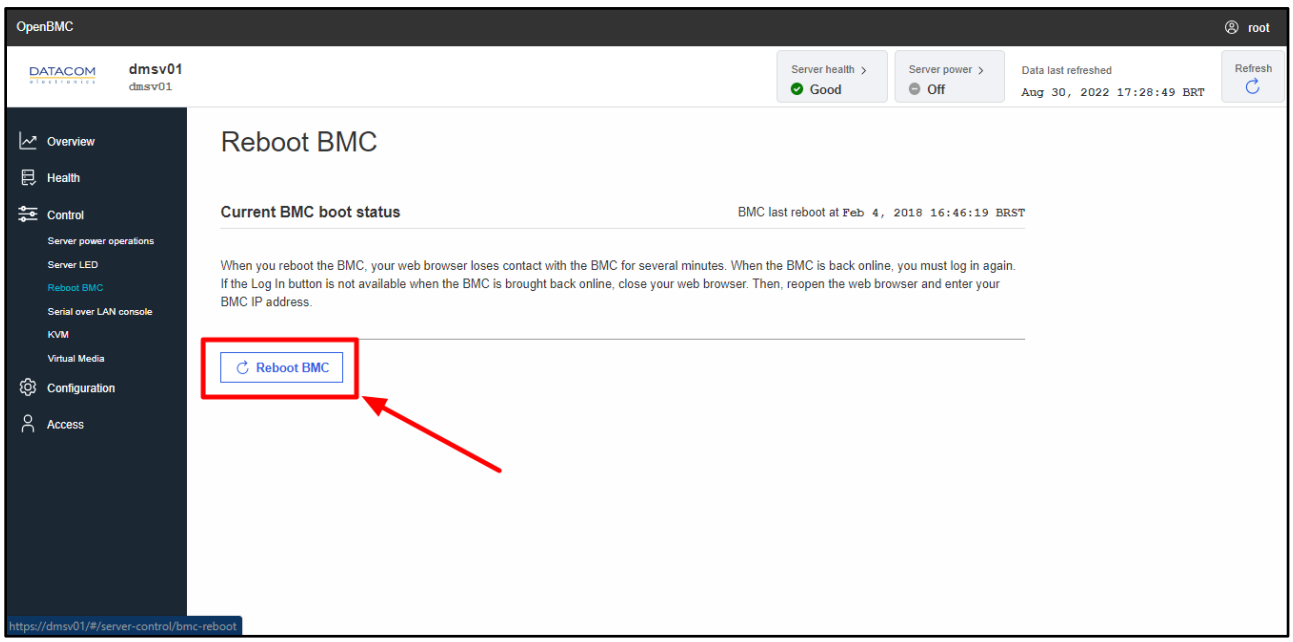


Figure 27: Reboot BMC menu

2.3.4 Serial over LAN console

The “Serial over LAN console” screen can be used to access the CPU console output directly in the BMC web management page.

The Figure 28 shows the SoL display when the BIOS/UEFI screen is being shown through the console.

Important: in order to allow the console screen to be shown, the BIOS/UEFI options “Console Redirection” must be enabled. For additional details regarding BIOS configuration please refer to the BIOS User Manual (2).

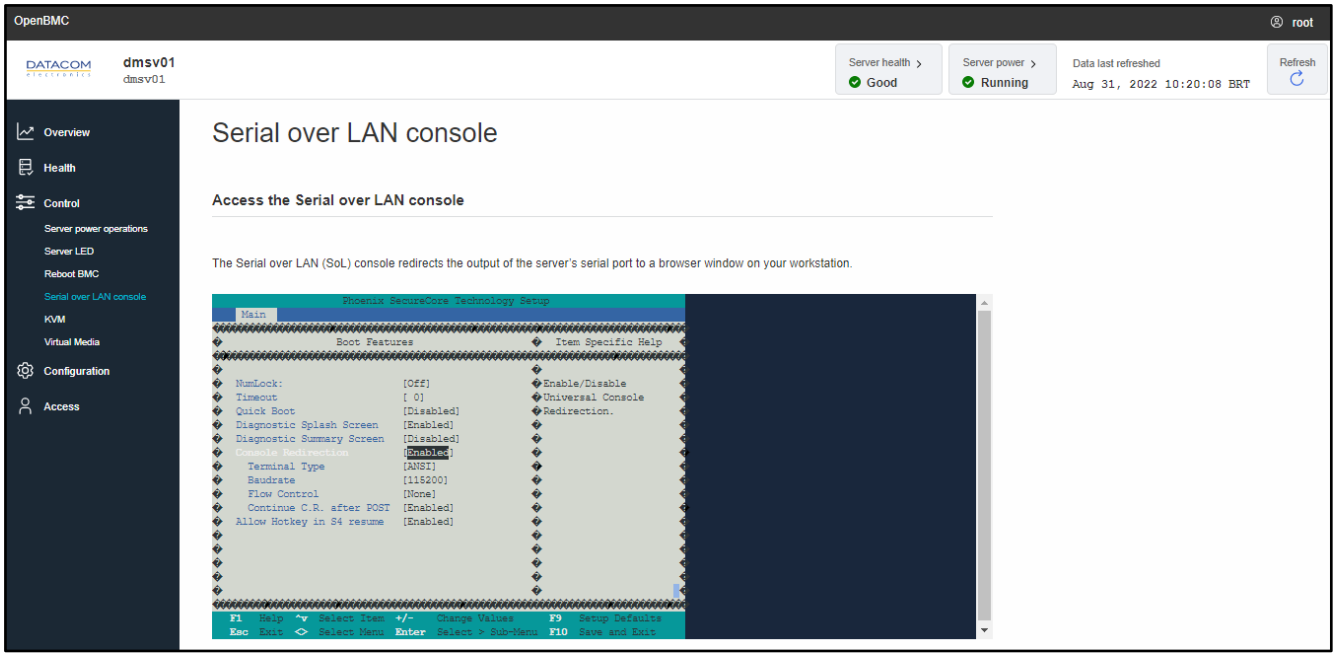


Figure 28: Serial over LAN (SoL) window

2.3.5 KVM

The KVM (Keyboard, Video and Mouse) console is a special window available in the BMC web GUI that provides to the user a real time access to the video output of the host, as well as to interact with the BIOS/UEFI or OS by means of the keyboard and mouse directly in the web page.

This allows full remote access to the server BIOS/UEFI and OS, once the users can perform any operation as if they have direct access to the equipment.

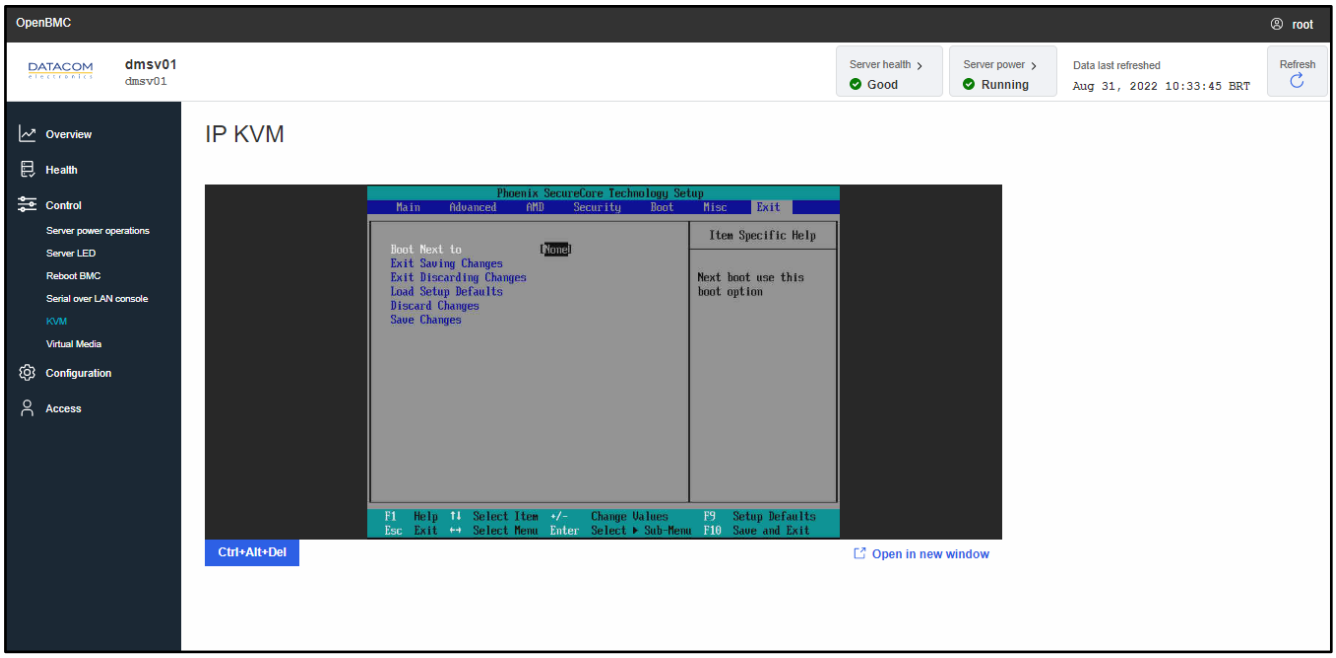


Figure 29: KVM window showing the BIOS/UEFI menu

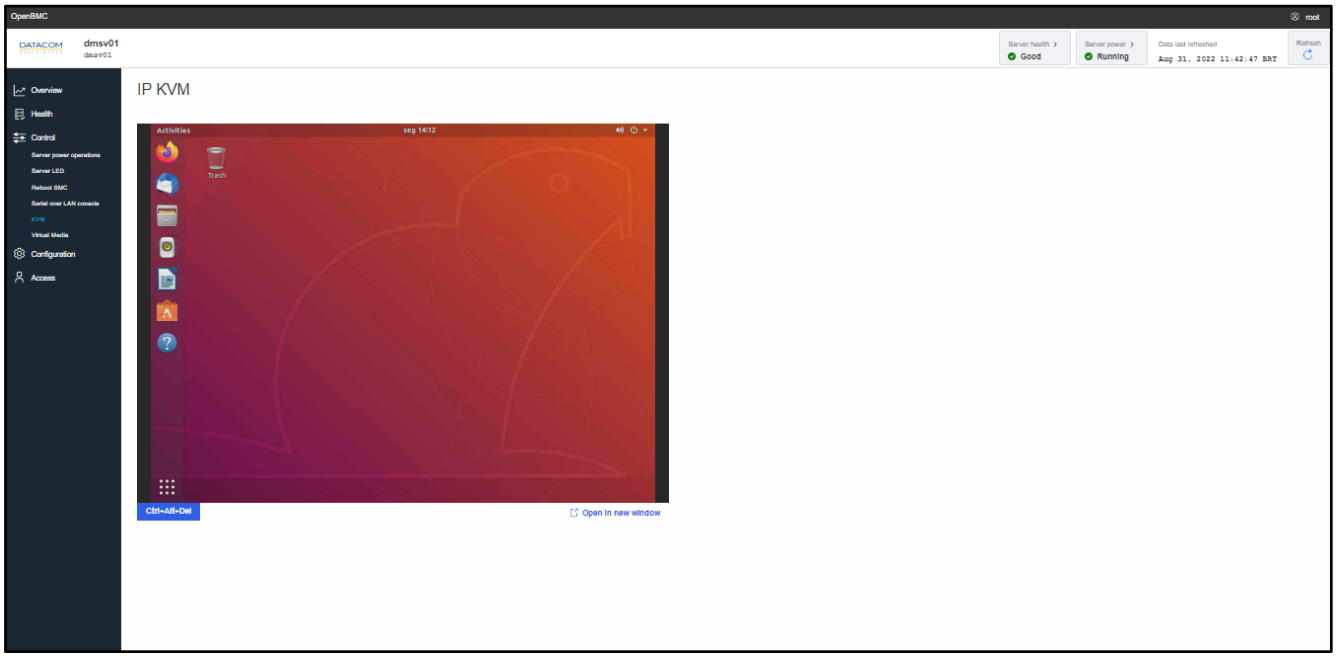


Figure 30: KVM window showing the OS menu

The KVM menu also has a “Ctrl+Alt+Del” button shortcut right below the window, that can be used whenever the user needs to send this command to the OS.

Additionally, there is the “Open in new window” button, which can be used to open a new instance of the web browser where the KVM screen will be displayed. This button does not close the current KVM screen being displayed, it just opens a new window in the web browser with another KVM screen.

2.3.6 Virtual Media

The “Virtual Media” menu is used to load and activate an “.ISO” file for installing or running operating systems. The procedure to activate the virtual media is the following:

1. Access the “Virtual Media” menu in the BMC web GUI and click “Choose file”. Then, select the “.ISO” image you would like to load.
2. Click start to activate the “.ISO” image in the system.

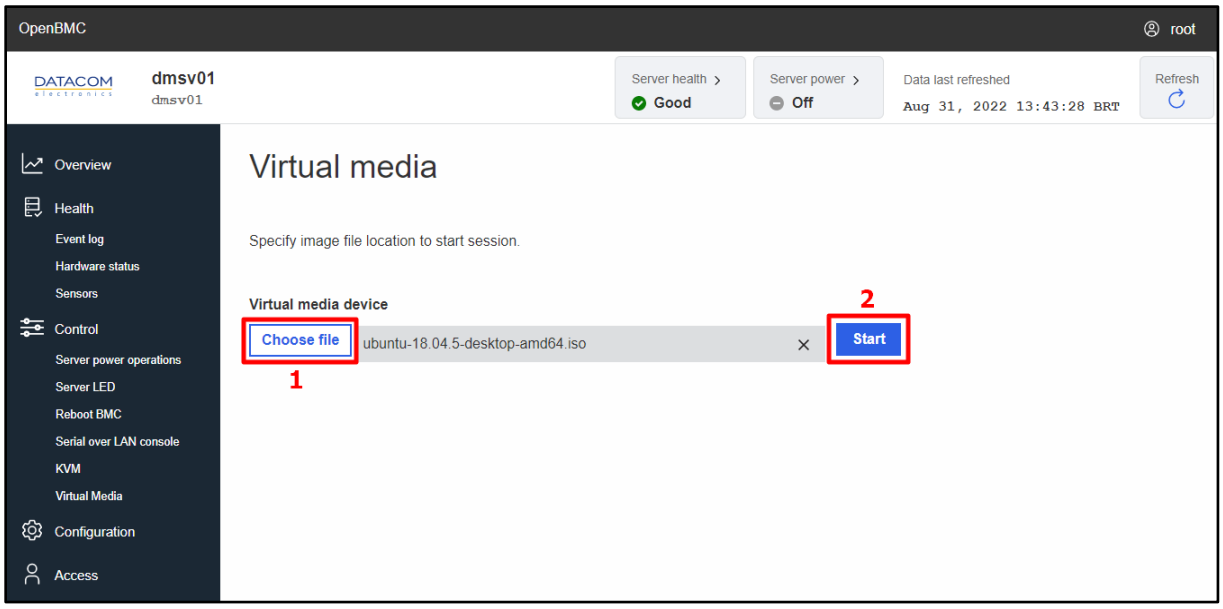


Figure 31: Virtual Media activation procedure

When the “.ISO” image is active, the text “Active Session” becomes available in the screen, as shown in the Figure 32.

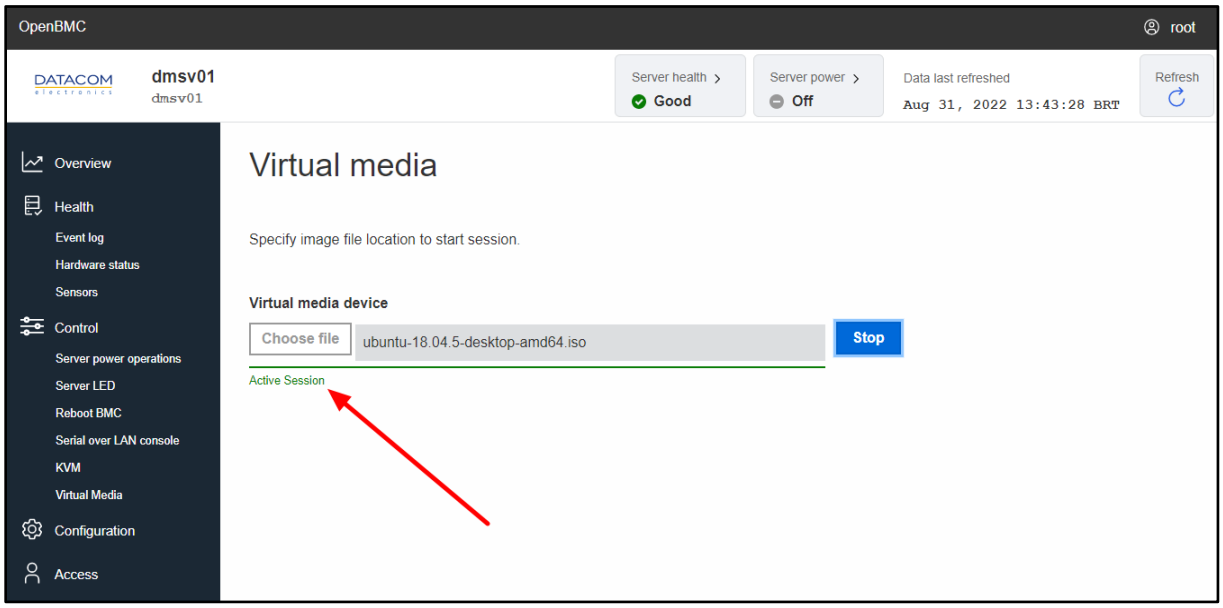


Figure 32: Virtual Media active

When the virtual media is active, it is possible to configure the server to boot from it as if it was a CD device. There are two means of selecting the ".ISO" media as the boot option:

1. Configuring the "CD" option in the "Boot Override" menu, as explained in section "2.3.1.2 Host OS boot settings - Boot override".
2. Accessing the BIOS/UEFI menu, and selecting the "CD" option in the Boot menu, as shown in Figure 33. When the virtual media is active, the text "Linux File-Stor Gadget" is shown together with the CD option.

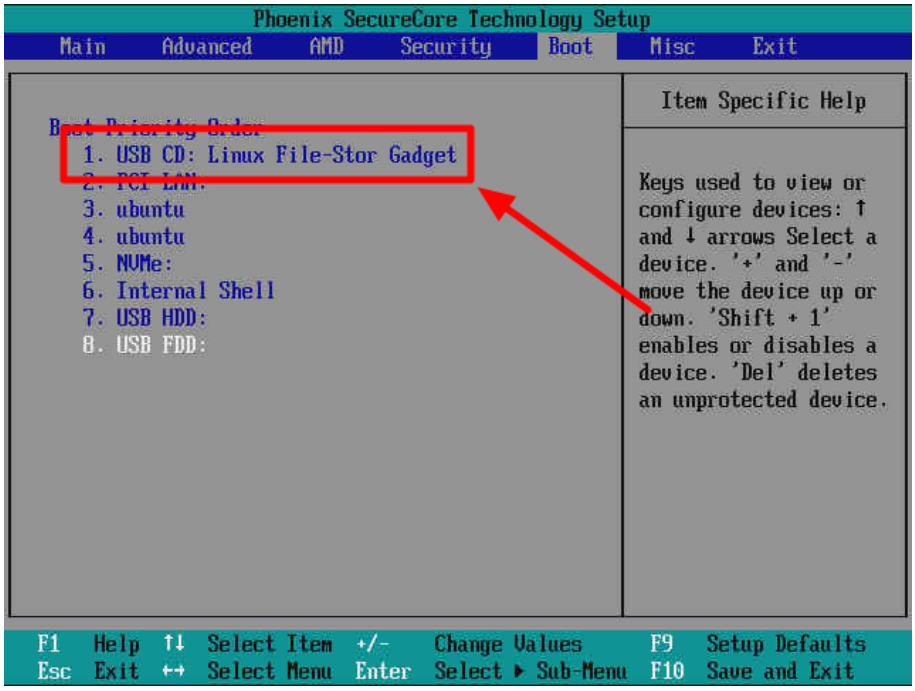


Figure 33: Boot Priority menu - virtual media option

2.4 Configuration Menu

2.4.1 Network settings

This option may be used to configure the network settings of the BMC.

2.4.1.1 Common Settings

There are two network interfaces available for configuration:

- **eth0:** this interface is the "NC-SI" (Network Controller Sideband Interface). The NC-SI interface is used for inband management of the BMC. The eth0 interface is accessed by means of the mezzanine card Ethernet port 0.
- **eth1:** it is the default out-of-band management interface of the BMC. It can be accessed by means of the dedicated Ethernet port present in the front panel of the DM-SV01.

For additional information regarding the BMC Ethernet ports and how to connect to them, please refer to the "DM-SV01 Product Manual" (1).

The user can select which of these interfaces is going to be configured by means of the drop down box shown in the Figure 34.

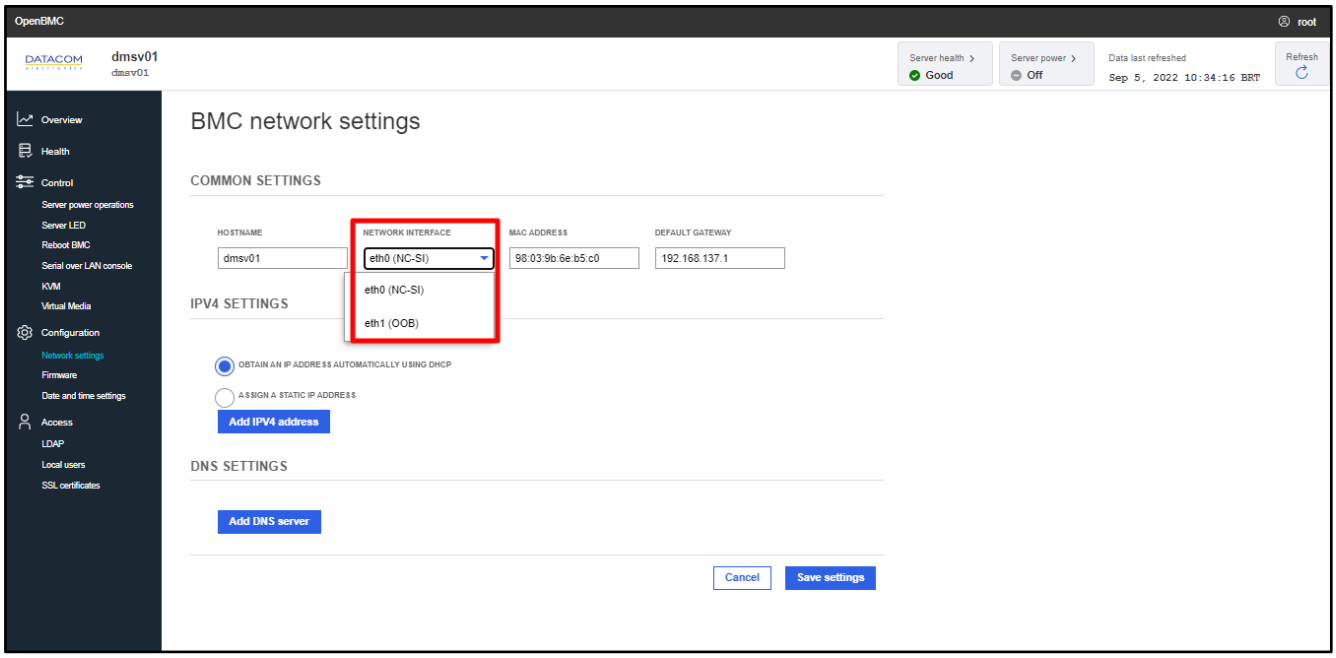


Figure 34: BMC network settings - selecting interface

By default, both eth0 and eth1 interfaces are configured as DHCP clients, so they can get their respective IP addresses automatically.

Besides selecting the network interface to be configured, the following common settings are available, which applies for both eth0 and eth1 interfaces:

- **Hostname:** configures the hostname of the BMC, which can be used to access it through the web browser or to connect through SSH.
- **Mac Address:** read only option, it is used to check the MAC address of the selected network interface.
- **Default gateway:** read only option, it is used to check the Default Gateway that is being used by the selected network interface.

2.4.1.2 IPV4 Settings

The settings from this section are individual for each Ethernet port, so the user must select the Ethernet interface eth0 or eth1 to be configured (as shown in section “2.4.1.1 Common Settings”) before proceeding. There are two options available for configuration:

- Configure the network interface as a DHCP client to receive the IP address automatically (default option).
- Assign a static IP address to the network interface.

The selection is done by means of the checkboxes shown below:

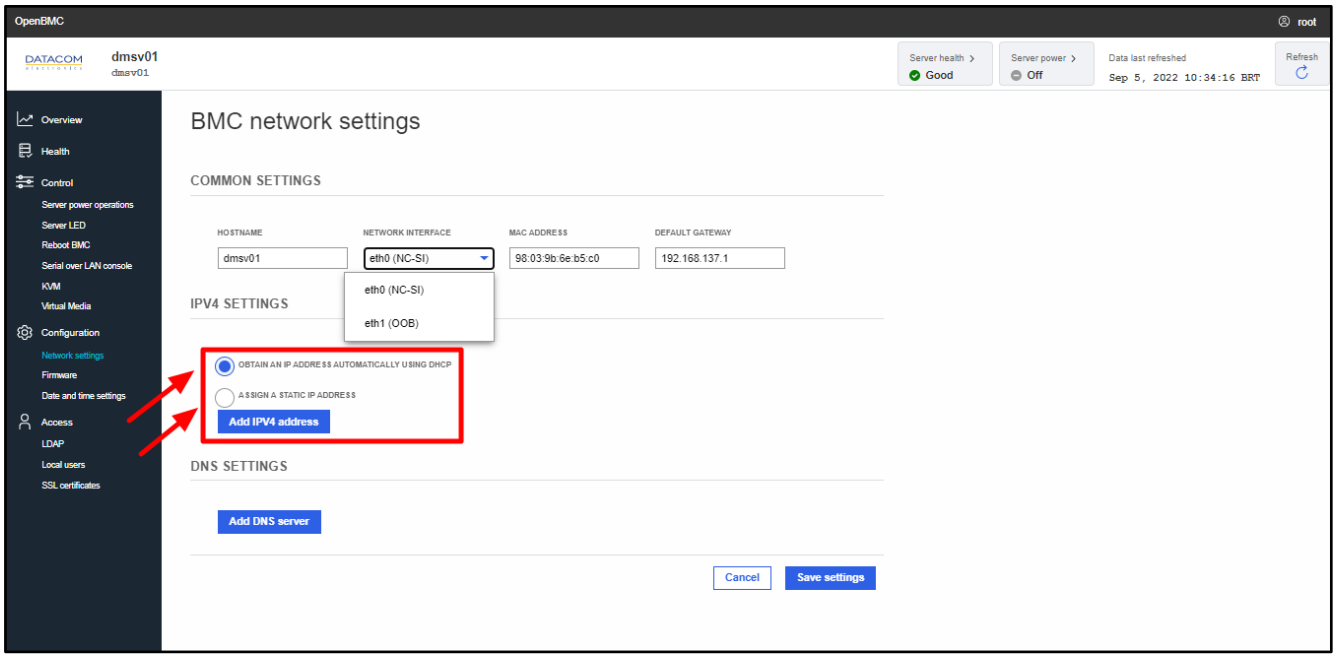


Figure 35: BMC network settings - configuring DHCP or static IPv4 settings

When the option “Assign a static IP address” is selected, the user is prompted to fulfill the following options:

- IPv4 address: in the format “111.111.111.111”.
- Gateway: in the format “111.111.111.111”.
- Netmask prefix length: integer number from “1” up to “32”.

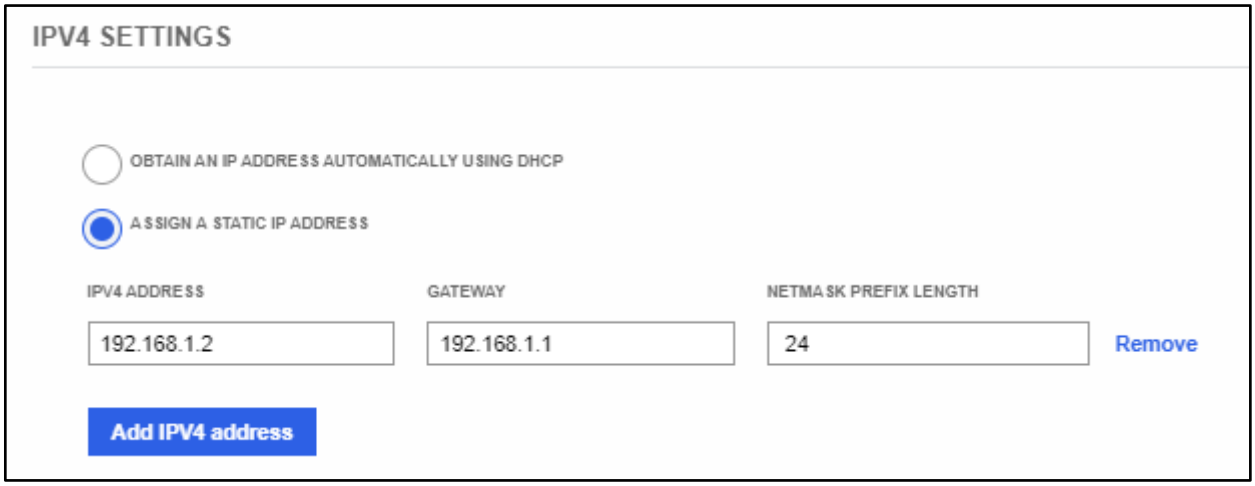


Figure 36: BMC network settings - configuring static IPv4 entries

The operation can be confirmed by clicking on the “Save Settings” button (step 3 in Figure 37).

2.4.1.3 DNS Settings

This option allows the user to manually configure one or more DNS (Domain Name Server) servers. The procedure for adding a DNS address is pretty straightforward. The user just needs to click on the “Add DNS server” button (step 1 in Figure 37) and then fill the IP address inside the desired text box (step 2 in Figure 37). The operation can be repeated for adding more DNS server addresses if needed. The operation can be confirmed by clicking on the “Save Settings” button (step 3 in Figure 37).

Figure 37: BMC network settings - configuring DNS server

2.4.2 Firmware

2.4.2.1 Current FW versions

In the “Firmware” menu, the user can view the current versions of BIOS and BMC firmwares, as well as to perform a FW update.

The image below shows an example of the FW menu section that shows the current FW versions of both BMC and BIOS running on the system.

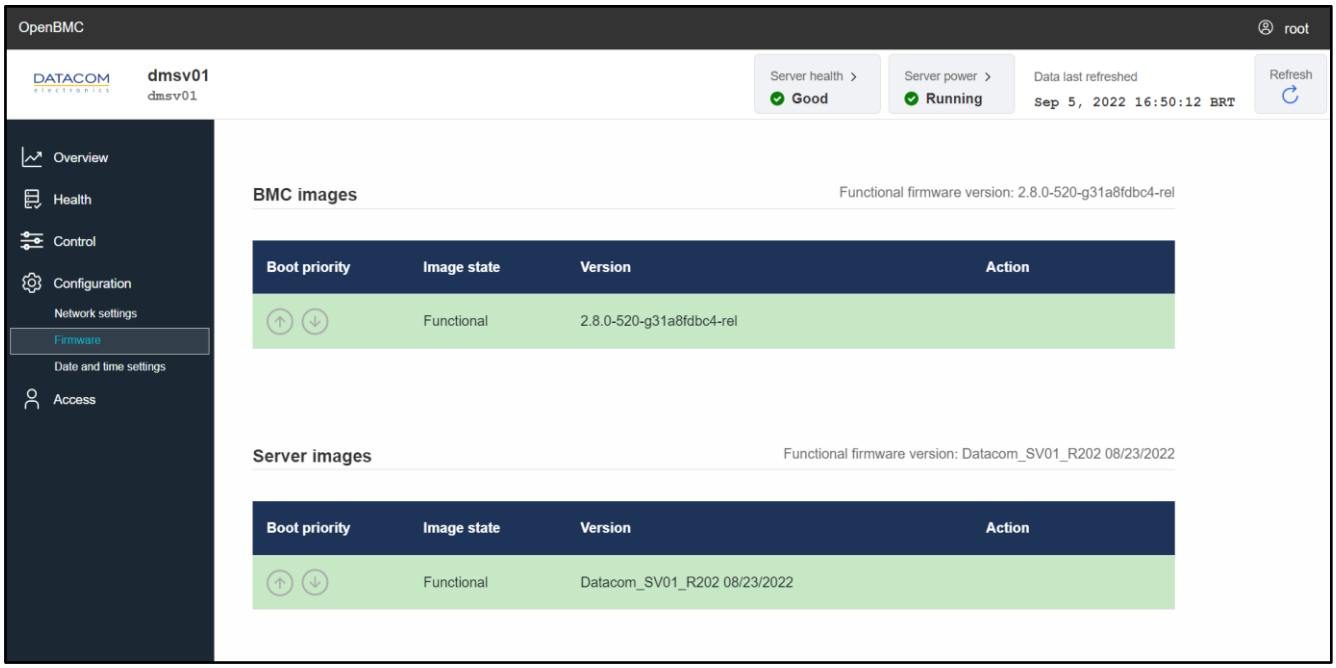


Figure 38: Firmware menu - current FW versions

2.4.2.2 FW update process - BMC or BIOS

The FW update process for both BMC and BIOS is very similar and it is composed of two main steps: FW image upload and FW activation.

2.4.2.2.1 FW image upload

The section “Specify image file location” is used to upload the BIOS or BMC image file for performing the FW update. There are two options available:

- **Option 1:** Upload image file from workstation: the file is uploaded by clicking on the “Choose a file” button, selecting the file image (BIOS or BMC) from the workstation and then clicking on “Upload firmware”.

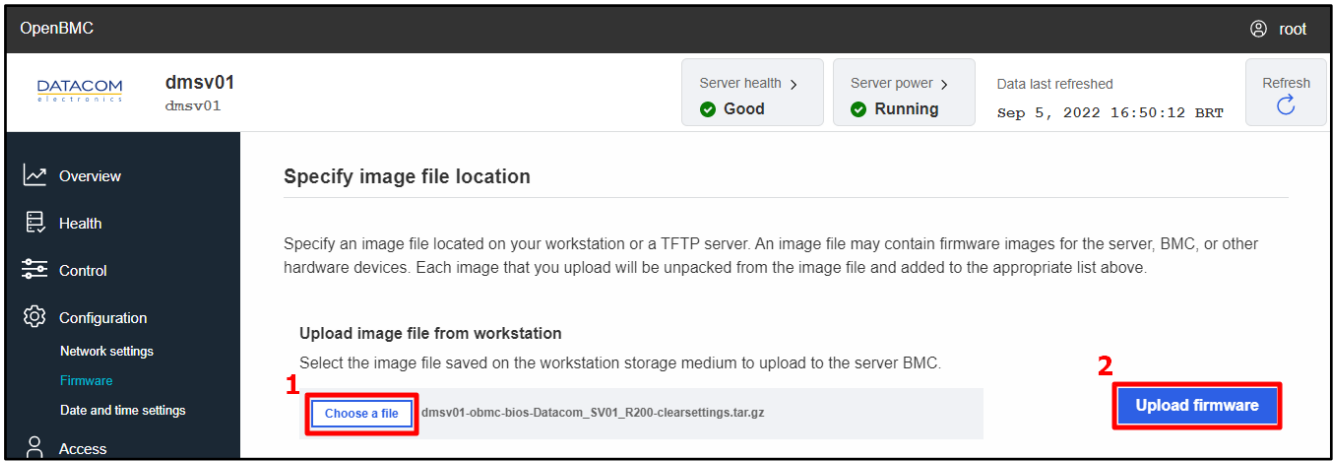


Figure 39: Firmware menu - choosing a file for performing FW update

- **Option 2:** Download image file from TFTP server: the user specifies the TFTP server IP address and the exact file name (BIOS or BMC) and then clicks on “Download firmware”.

Figure 40: Firmware menu - configuring TFTP server for file transfer

Once the file is uploaded using any of the means described above, the system automatically detects the image type (BIOS or BMC) and prepares it for the update.

The BMC checks if the image is valid and signed and pops up a message in the right top corner of the screen indicating if the upload is accepted or not. Once the image is uploaded and properly accepted, the information about the new FW image is placed in the respective image field. The Figure 43 shows an example of a BIOS FW image already uploaded and ready for the update.

The BIOS FW has two types of image for updating:

- **Keep BIOS settings image:** this image updates the BIOS FW, but keeps the current settings unchanged. All the configurations performed by the user will be preserved after the update. This type of update will be available only for minor FW upgrades, when it is safe to keep the settings unchanged during the update.
- **Clear BIOS settings image:** this image updates the BIOS FW and resets all BIOS settings to the factory default. The changes performed by the user will not be preserved after the update.

Please consult the Datacom sales team whenever necessary for checking which type of image is available for the update.

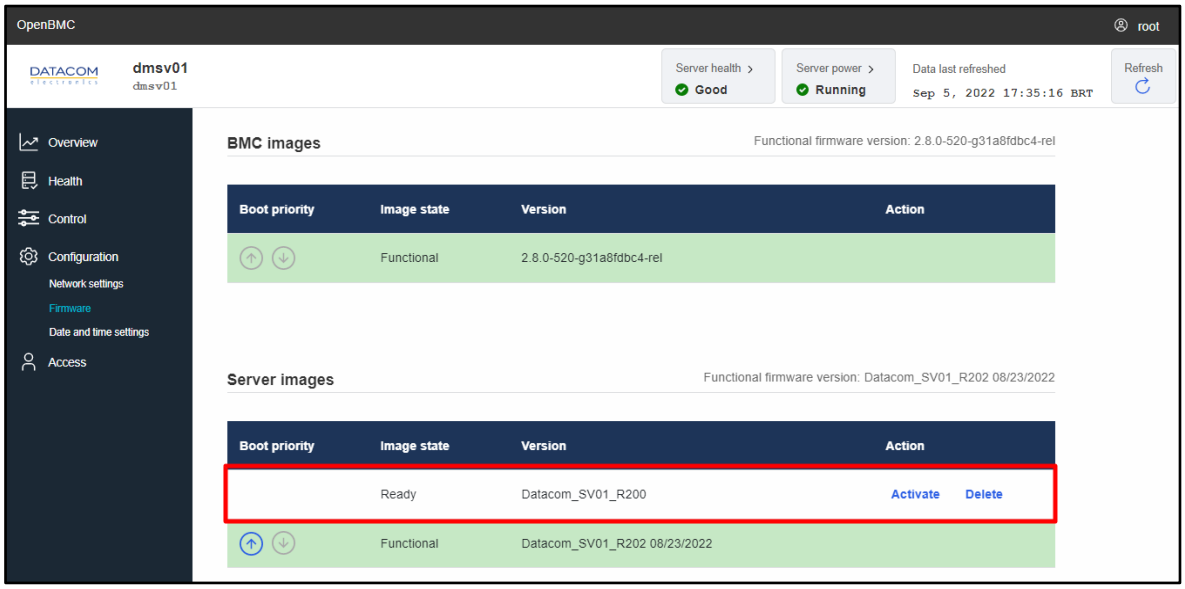


Figure 41: BIOS FW ready for the update

The Figure 42 shows an example of a BMC FW image already uploaded and ready for the update.

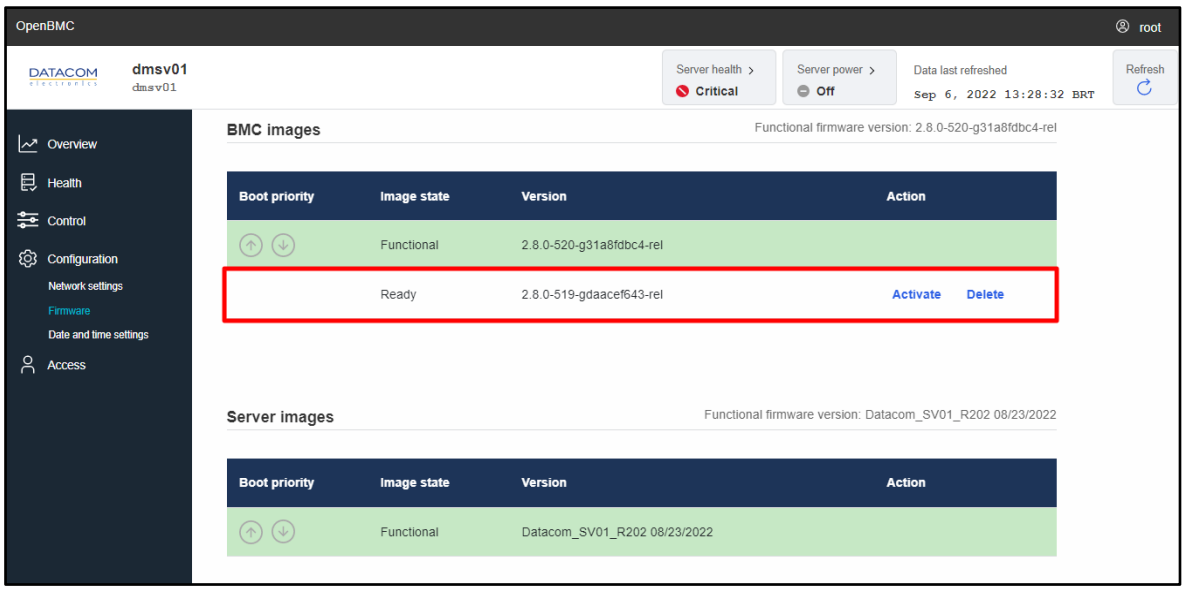


Figure 42: BIOS FW ready for the update

2.4.2.2.2 FW activation

After uploading the FW image by following the steps described in section “2.4.2.2.1 FW image upload”, the user can start the firmware update process by clicking on the “Activate” button.

The Figure 43 shows an example of FW update, where a BIOS image has been successfully uploaded and is ready to be activated.

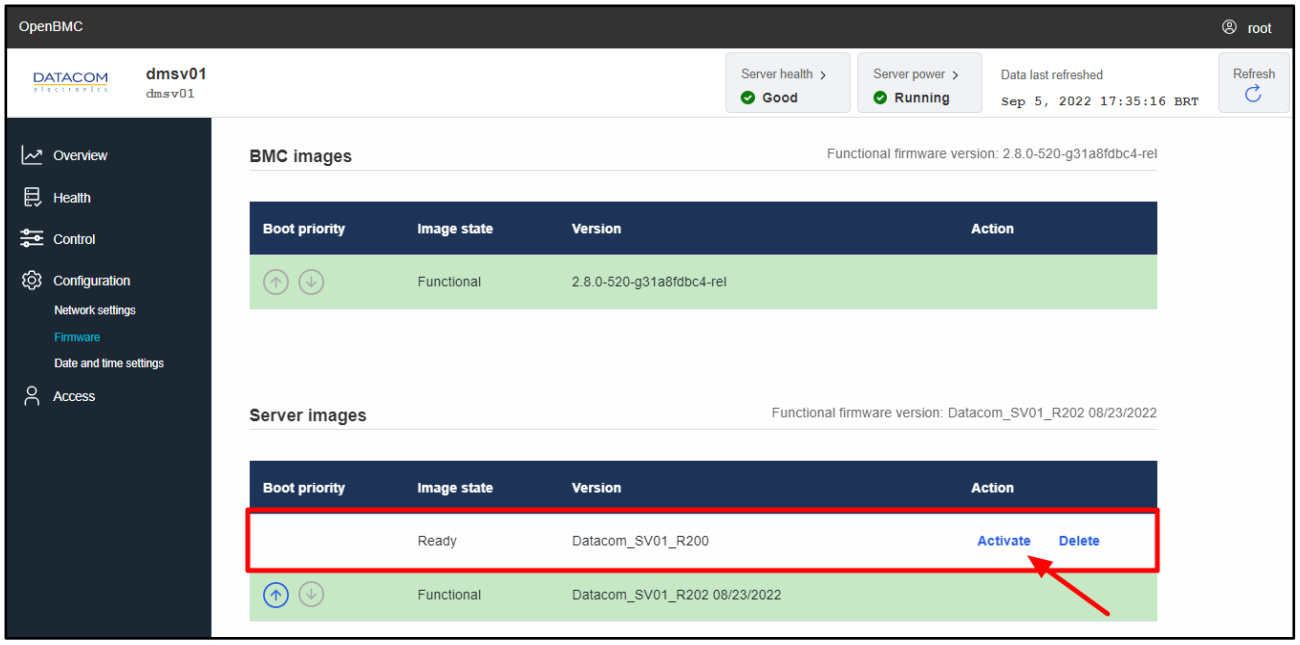


Figure 43: BIOS FW activation

The Figure 44 shows an example of FW update, where a BMC image has been successfully uploaded and is ready to be activated.

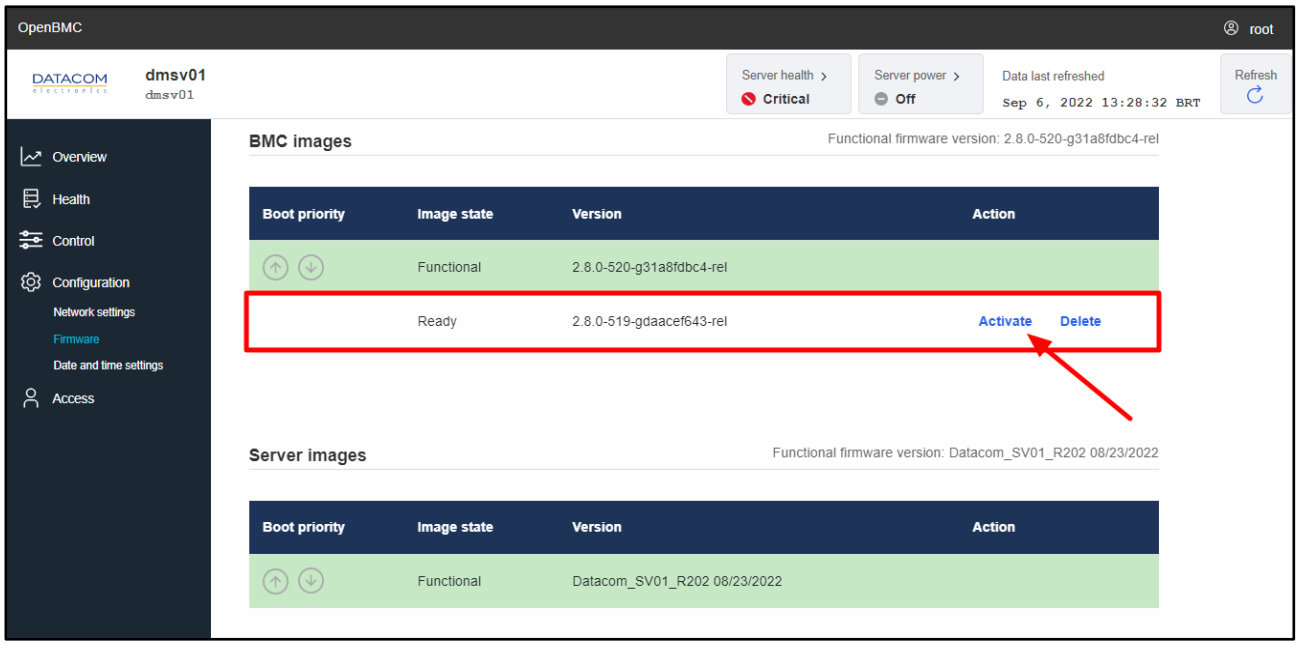


Figure 44: BMC FW activation

When clicking on the “Activate” button, a confirmation message is displayed on the screen.

In case of a BIOS update, the user may choose between two options:

- **Activate Firmware File Without Rebooting Server:** the FW image will be updated, but the host CPUs will not be rebooted. When this option is selected, any BIOS FW change will not take effect until the next reboot is performed.
- **Activate Firmware File and Automatically Reboot Server (recommended):** the FW image will be updated and the host CPUs will be rebooted automatically. When this option is selected, any BIOS FW change will take effect immediately after the automatic reboot is performed.

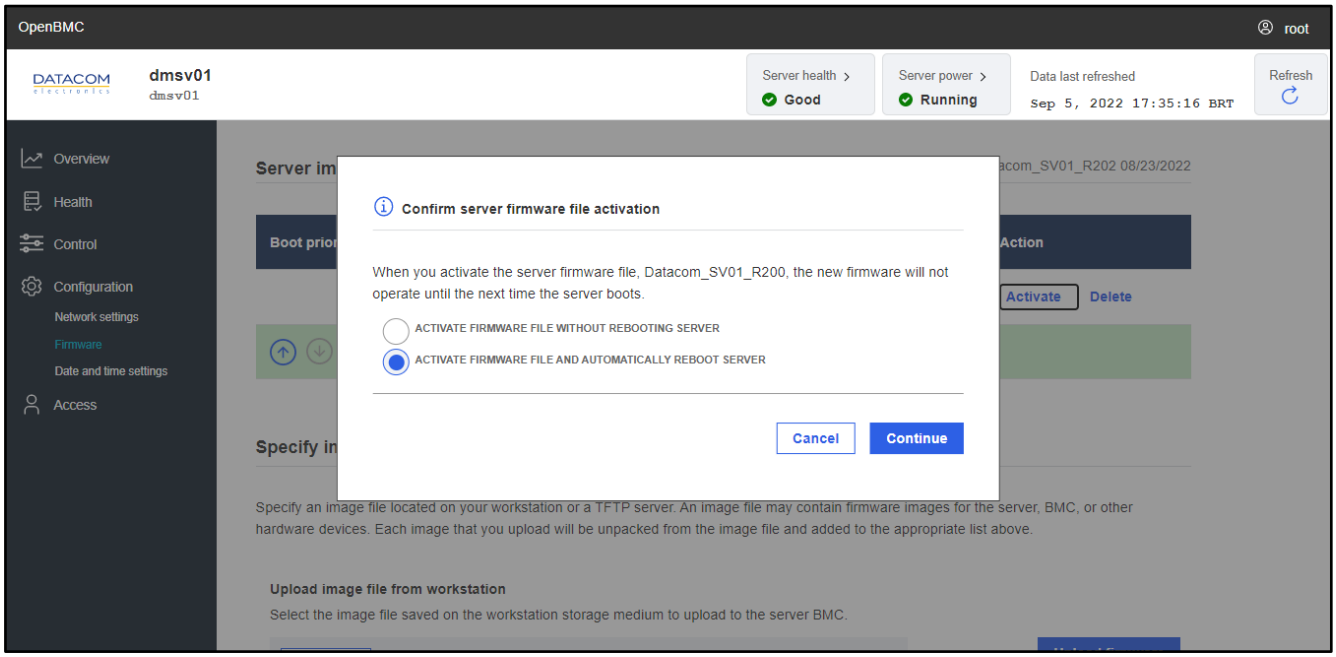


Figure 45: BIOS FW activation confirmation

In case of a BMC update, the user may choose between two options:

- **Activate Firmware File Without Rebooting BMC:** the FW image will be updated, but the BMC will not be rebooted. When this option is selected, any BMC FW change will not take effect until the next BMC reboot is performed.
- **Activate Firmware File and Automatically Reboot BMC (recommended):** the FW image will be updated and the BMC will be rebooted automatically. When this option is selected, any BMC FW change will take effect immediately after the automatic reboot is performed.

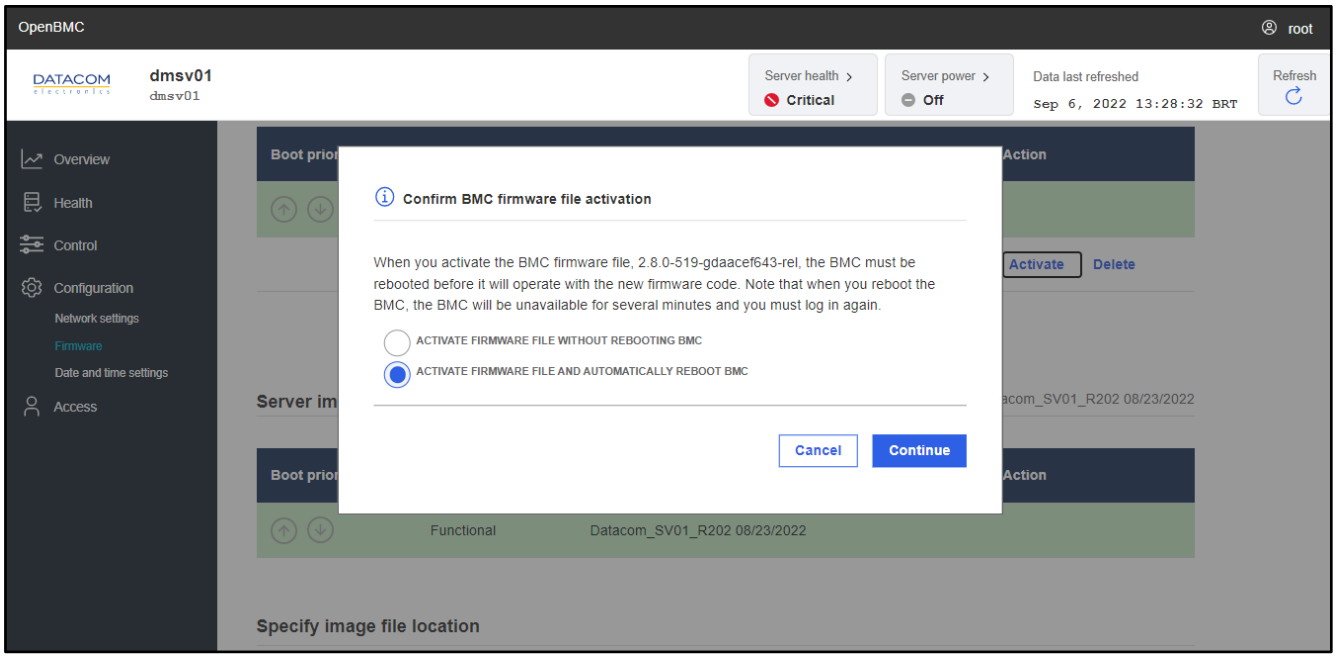


Figure 46: BMC FW activation confirmation

Important: The BMC reboot process does not interfere with the host CPUs operation. The BMC operation is independent of the host CPUs power, so the BMC can be safely rebooted while the host CPUs keeps processing the workloads.

Important: The user will lose access to the BMC web management GUI while the BMC reboot is being performed, but the access will be restored as soon as the BMC reboot is completed. All the network settings will be preserved during the BMC FW update process, so the user will keep having access to the BMC after the reboot.

2.4.2.3 Factory Reset - BIOS and BMC

In the “Firmware” menu, there are also two additional buttons that are used to reset BIOS and BMC settings to factory default. These options can only be set by users with administrator privileges.

- **BIOS Settings Reset:** this option returns all the BIOS settings to the factory default. Any change performed in the BIOS menus will be reverted back to the default value when using this function. Please note that some settings may require a host reset or power cycle to be applied. Additional details can be found in the DM-SV01 BIOS User Manual (2).
- **BMC Factory Reset:** this option resets all BMC settings to the factory default.

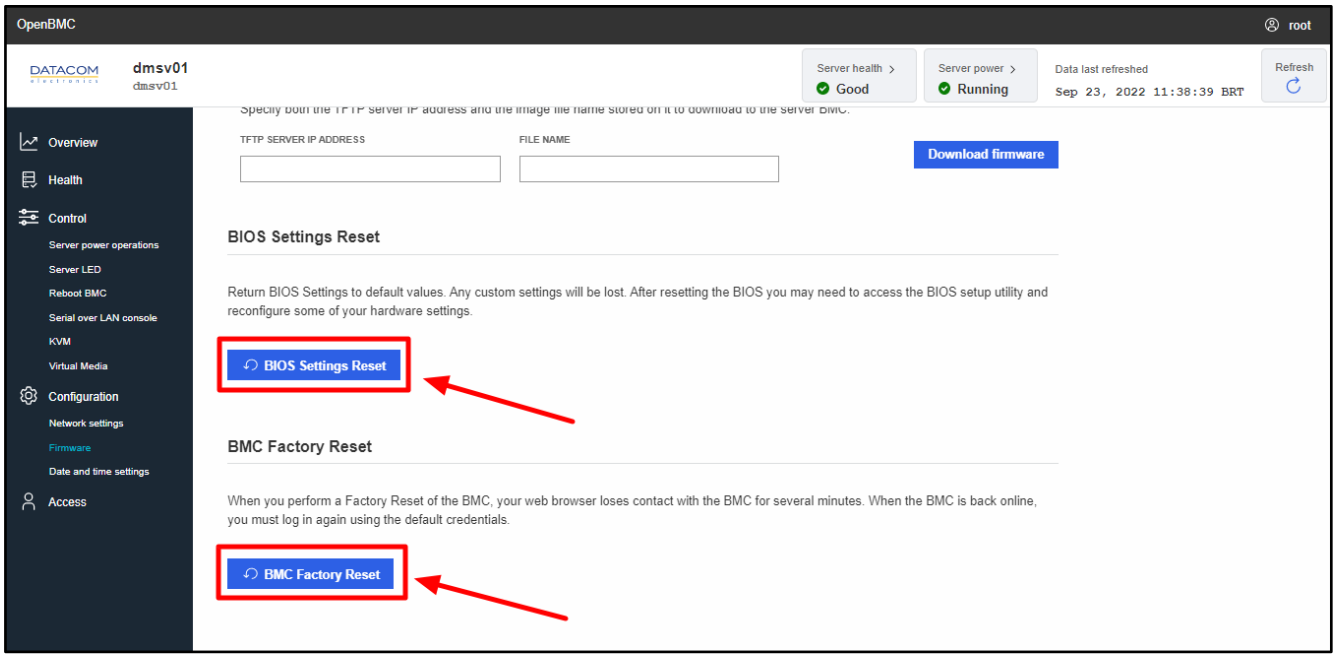


Figure 47: BIOS and BMC factory reset

Important: care should be taken before applying the factory reset to the BMC. The user will lose access to the BMC for some minutes when this action is taken, and all the BMC settings will be reset to their default values, including the users and network settings. Therefore, after the factory reset is complete, the user will be able to access the BMC by sending the IP address through a DHCP server and logging in using the default credentials:

- User: root
- Password: 0penBmc (the first digit is the number “zero”)

2.4.3 Date and time settings

This menu allows the user to choose between two options for configuring the date and time:

- **Option 1:** configuring a NTP server to automatically obtain the date and time information. The steps to configure the NTP server are as follows:
  - 1) Mark the option “Obtain Automatically From a Network Time Protocol (NTP) Server”.
  - 2) Click on “Add new NTP server”.
  - 3) Fill in the text box with the NTP server IP address, in the format “111.111.111.111”.
  - 4) Click on the “Save settings” button below.

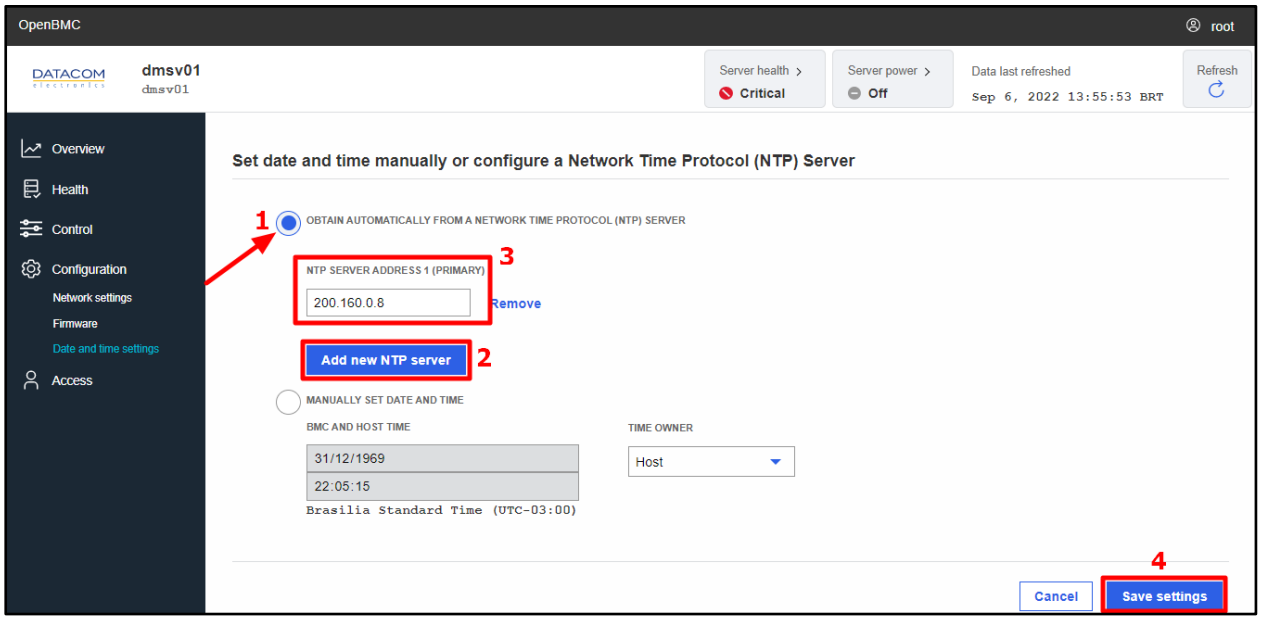


Figure 48: Date and Time configuration - configuring NTP server

- **Option 2:** configuring the date and time information manually:
  - **1)** Mark the option “Manually Set Date and Time”.
  - **2)** Fill in the text boxes with the date (in the format “dd/mm/yyyy”) and time (in the format “hh:mm:ss”) information.
  - **3)** Selects the time owner of the date and time setting. The time owner can be the BMC, the host, both or “split”.

    Note: If “split” is selected, the user is prompted to configure separately the BMC date and time and the BIOS date and time.

    Note: The owner option is only available in previous releases of BMC SW.
  - **4)** Click on the “Save settings” button below.

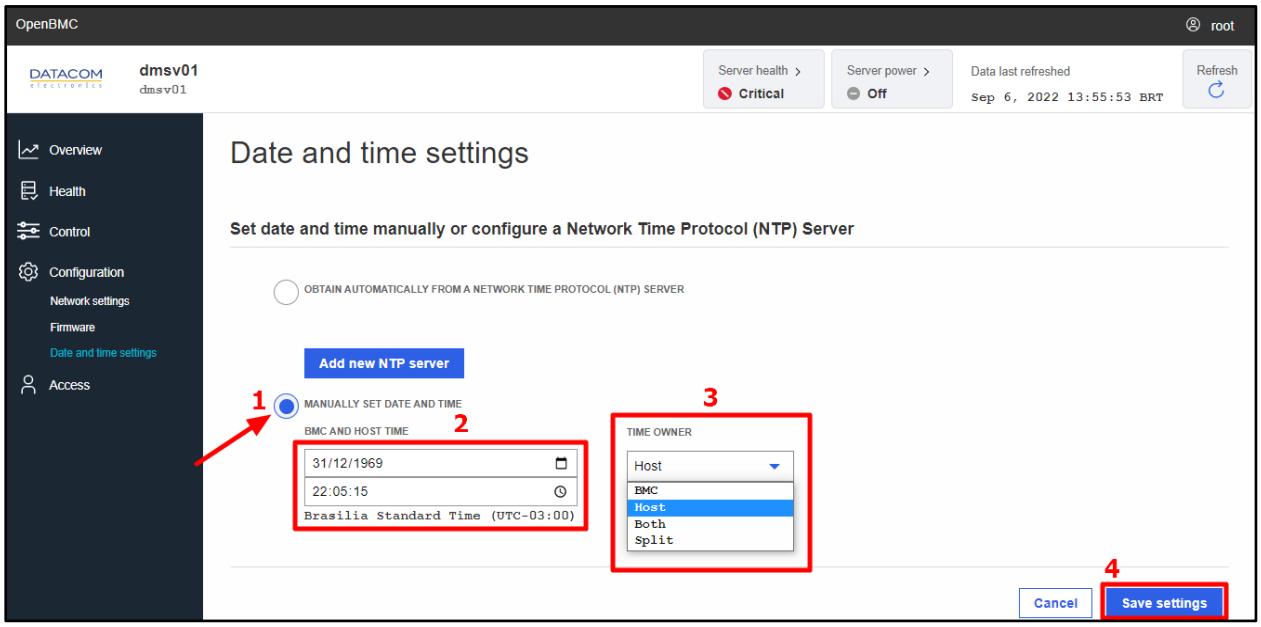


Figure 49: Date and Time configuration - configuring manually

2.5 Access Menu

2.5.1 LDAP

The LDAP menu is used to configure the basic LDAP settings and also to manage the role groups.

2.5.1.1 Enabling and configuring the LDAP

In order to configure the LDAP, the user must initially enable the LDAP authentication by marking the checkbox shown in the Figure 50.

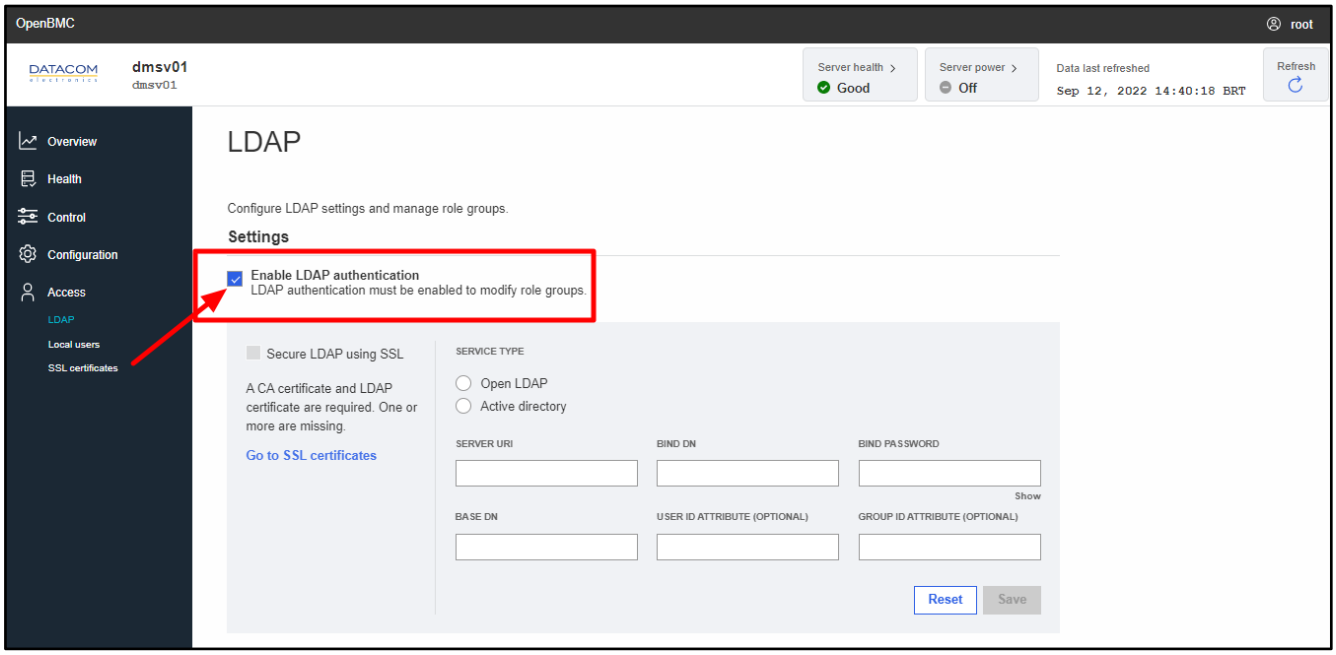


Figure 50: LDAP enable

Once enabled, the LDAP settings below are available for configuration:

- **Service Type:** the user must choose which service is running the LDAP - “Open LDAP” or “Active Directory”.
- **Server URI:** the user must specify the URI (Uniform Resource Identifier) to access the server, starting with the scheme “ldap://”. As an example, a valid entry could be “ldap://mycompany.com”.
- **Bind DN:** the bind DN (Distinguished Name) of the user authenticating to the LDAP directory. As an example, if the username is “user1” and it is part of the “Users” group, the entry should look like this: “CN=user1,OU=Users,DC=mycompany,DC=com”.
- **Bind Password:** the password related to the bind DN above
- **Base DN:** the base DN (Distinguished Name) of the user authenticating to the LDAP directory. An example of a valid entry should look like this: “DC=mycompany,DC=com”.
- **User ID Attribute (optional):** additional user ID attribute (CN), if applicable.
- **Group ID Attribute (optional):** additional group ID attribute (OU), if applicable.
- **Secure LDAP using SSL:** enables or disables the secure LDAP. In order to be able to enable the secure LDAP, the user must first create a certificate. The procedure to do so is described in the section “2.5.3 SSL certificates”.

The image below shows an example of the LDAP configuration already completed. The user can confirm the configuration by clicking on the “Save” button.

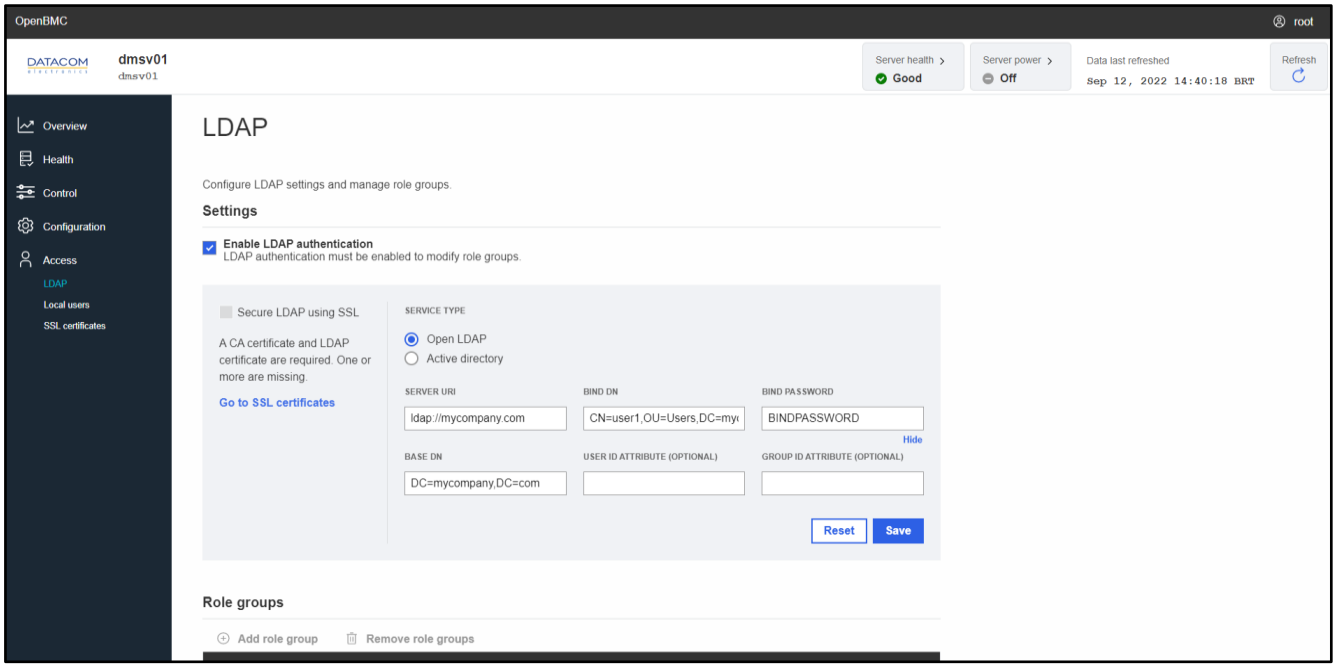


Figure 51: LDAP Configuration ready

2.5.1.2 Role Groups Management

Once the LDAP is properly configured, the user is now able to add role groups, by clicking on the “Add role group” button. Once the user clicks on this button, the configuration screen is shown and the user is prompted to set the following parameters:

- **Role Group Name:** the user must specify the name of the role group, as a string.
- **Privilege:** the privilege level for the group of users. There are four options available:
  - Administrator
  - Operator
  - ReadOnly
  - NoAccess

It is possible to check the actions allowed for each privilege level in the “Local Users” menu. Please refer to the section “2.5.2 Local users” for additional information about this.

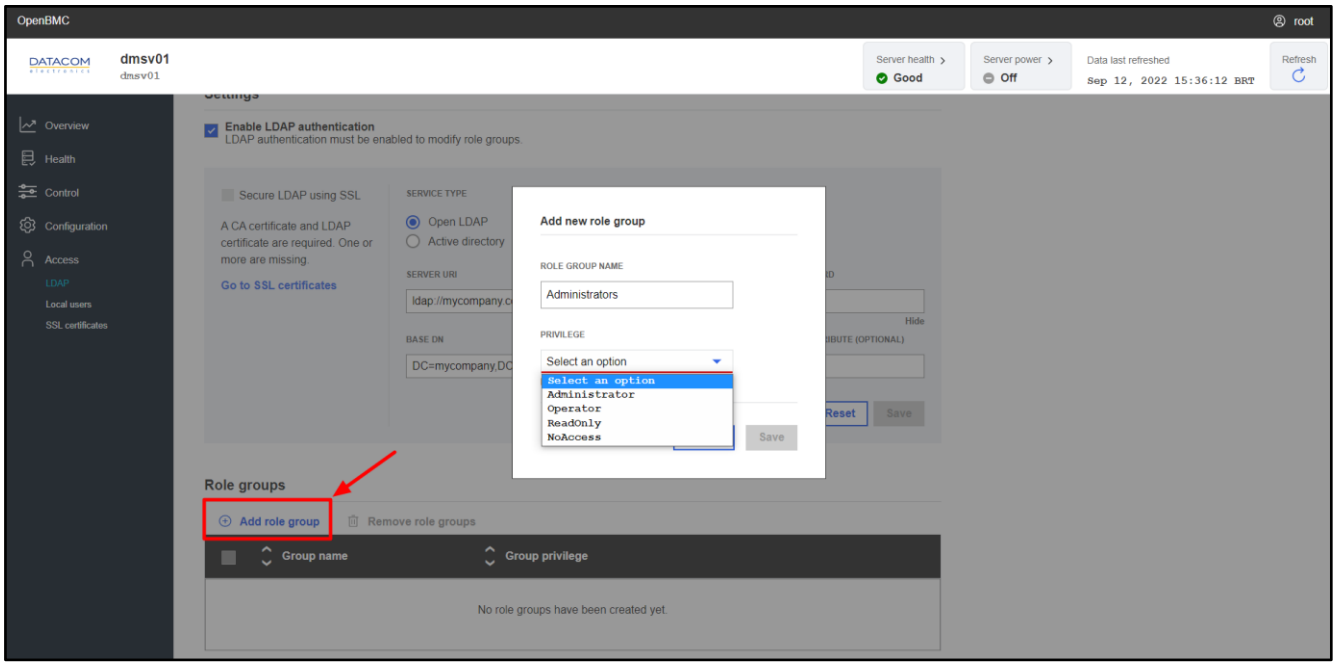


Figure 52: LDAP - Adding role group

The user can also edit or delete the previously created user groups, by clicking on the “pencil” button (edit) or “trash” button (delete).

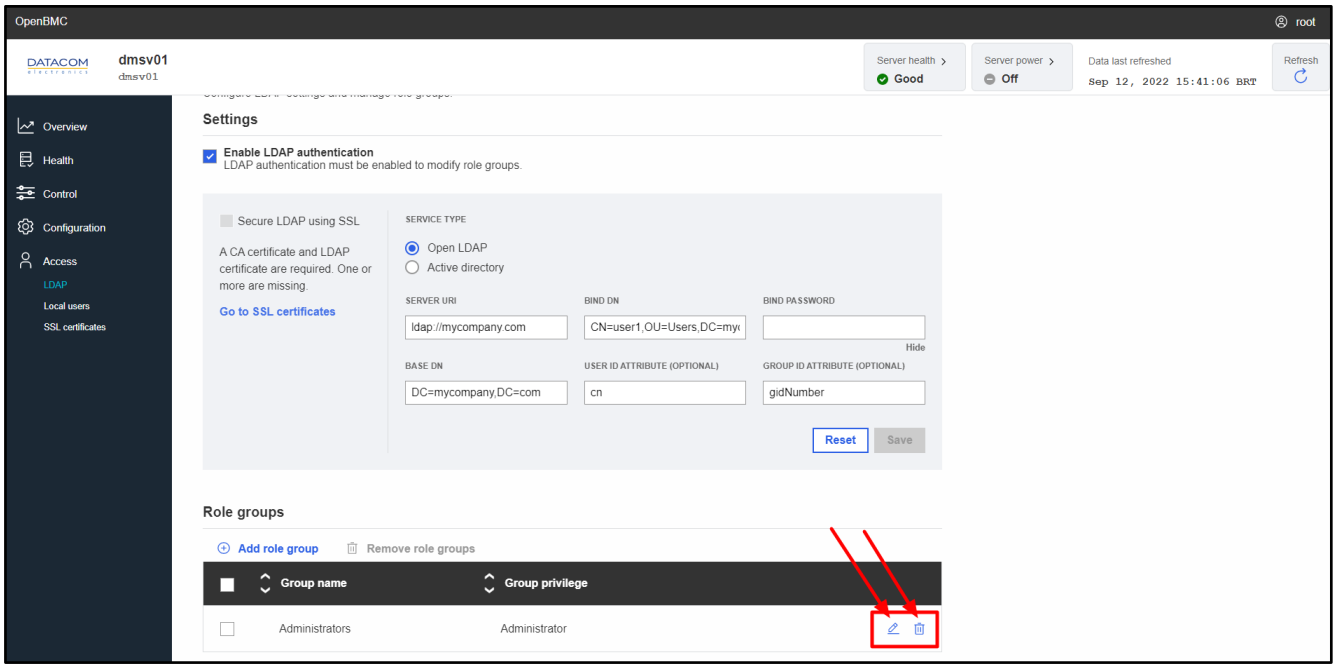


Figure 53: LDAP - Edit or delete user groups

2.5.1.3 Instructions for implementing the LDAP server

There are some recommendations that must be followed when implementing an LDAP server using OpenLDAP or Active Directory, in order to guarantee that the authentication will be performed successfully with the BMC:

1. The user must create a group in the LDAP server with the name “redfish”.
2. In the “redfish” group created in step “1”, the user must set the value “1004” for the “GidNumber” attribute.
3. Set the value “1004” to the “GidNumber” attribute of the user that will access the BMC through the LDAP.
4. The LDAP user must have the attribute “posixAccount” configured as an “objectClass”.
5. In the BMC web GUI, the user must follow the procedures defined in section 2.5.1.2 Role Groups Management and create a group with the name “redfish”, configured with administrator privileges.

2.5.2 Local users

The “Local users” menu allows the server administrator to create, delete and configure the access privilege level for each user. The Figure 54 shows the main screen of the menu.

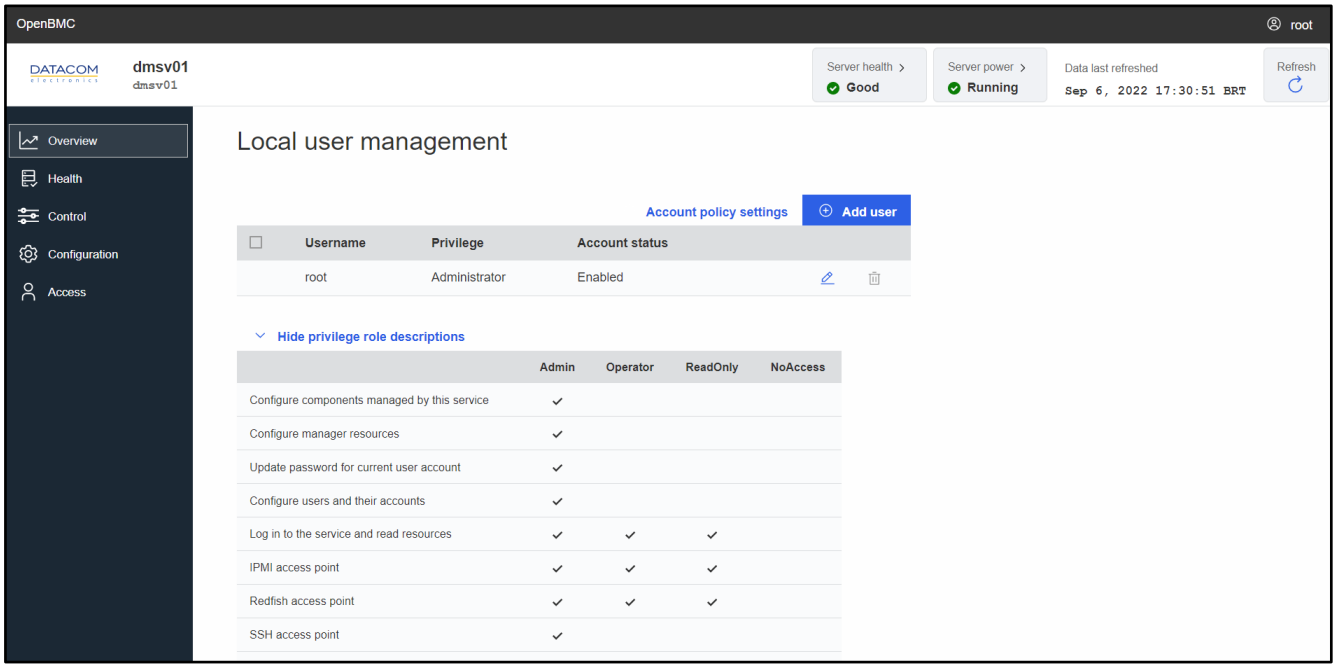


Figure 54: Local users menu

2.5.2.1 Account policy settings

The “Account policy settings” button is used to configure some basic settings related to security, as shown in the Figure 55.

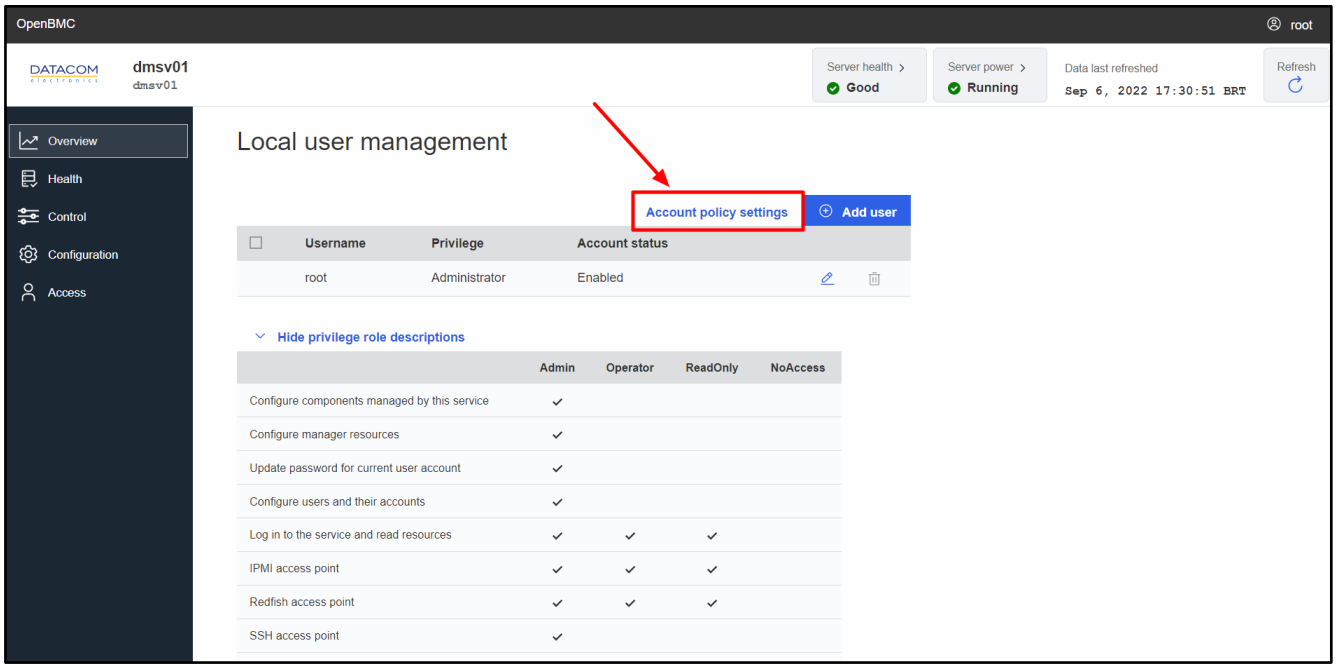


Figure 55: Account policy settings menu

The following options are available in the Account policy settings menu:

- **Max Failed login Attempts:** configures the maximum number of attempts allowed for trying to log in the BMC web GUI. If the number of attempts is exceeded, then the specific action defined in the “User Unlock Method” is triggered.
- **User Unlock Method:** when the maximum number of attempts for logging in the BMC web GUI is exceeded, the specific setting defined in this option is triggered. There are two possibilities for configuration:
    - **Automatic after timeout:** when this option is selected, the system will block the login and start a timer. The login attempt will be released again only after the time configured in the “Timeout Duration” window is reached. The “Timeout Duration” is configured in seconds in the text box shown in the Figure 56.
    - **Manual:** when this option is selected, the user is completely blocked whenever the maximum number of login attempts is exceeded.

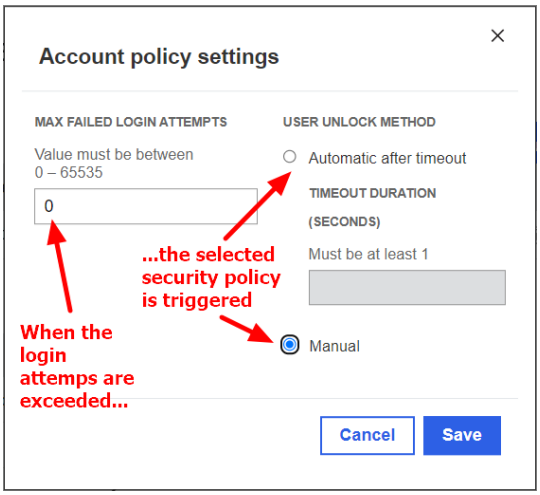


Figure 56: Account policy settings menu

When the user is blocked by the “Manual” policy, the only way to unlock it is by accessing the BMC with an alternative Administrator account, clicking on the pencil icon to modify the locked user and then clicking on the “Unlock” button, as shown in the Figure 58.

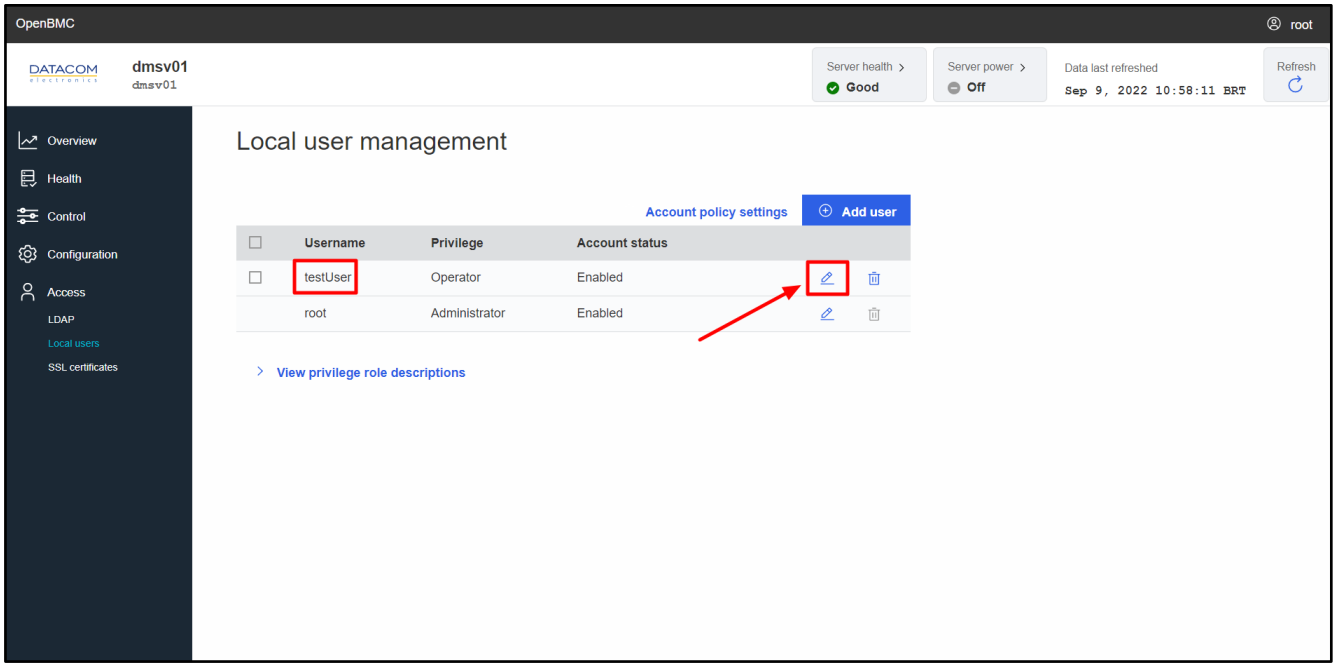


Figure 57: Edit User option

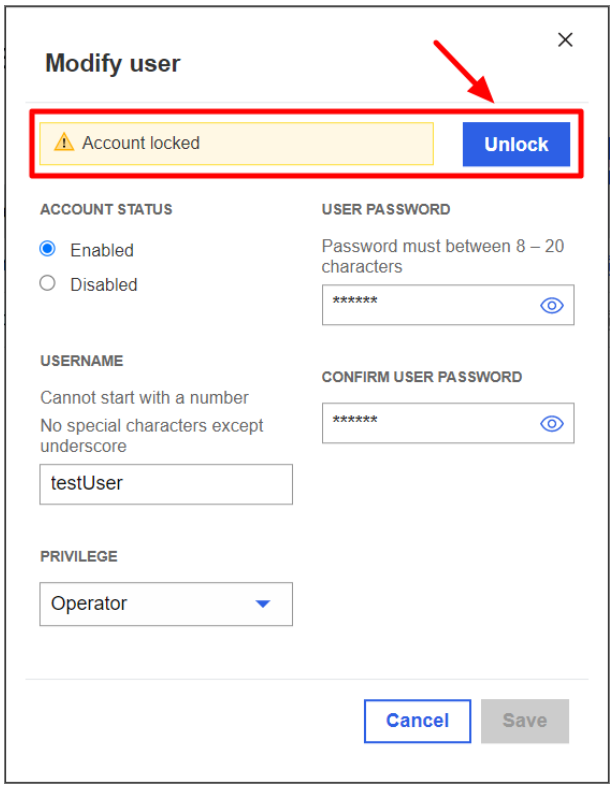


Figure 58: Unlock User button

2.5.2.2 Managing users

The “Add user” button can be used to create a new user for accessing and administering the BMC. The button is shown in the Figure 59.

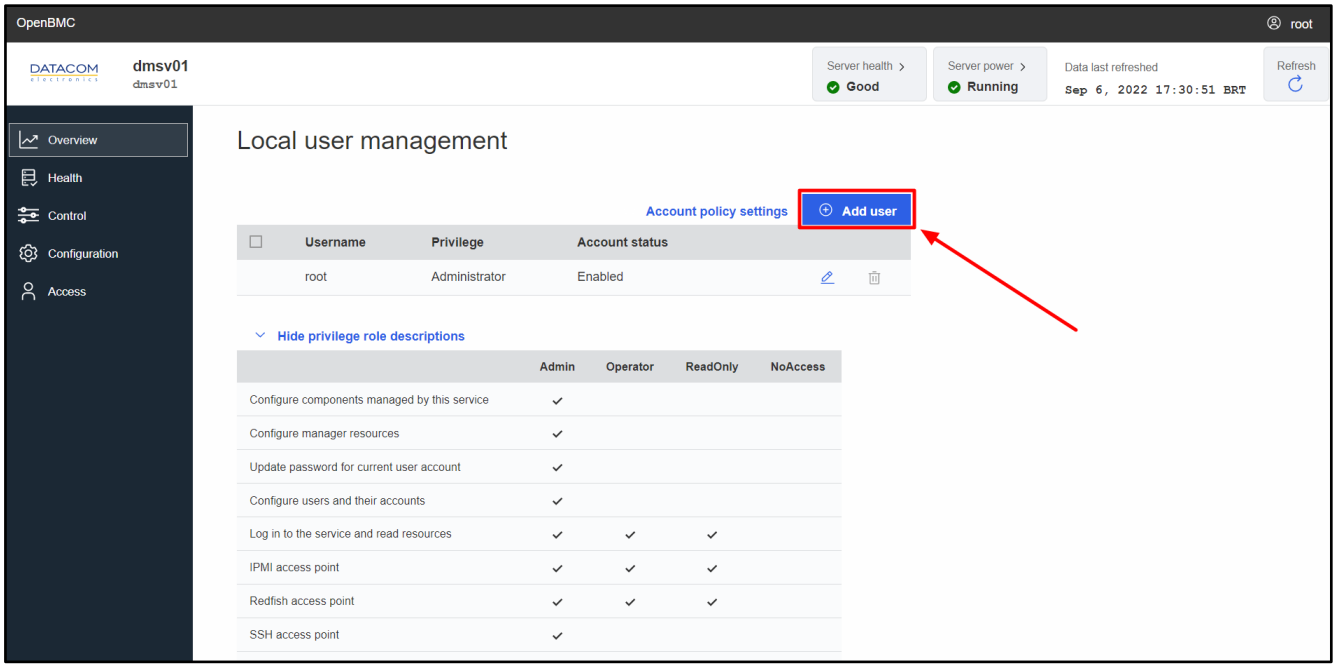


Figure 59: Add user button

The “Add user” window is shown in the Figure 60, and the following options are available for configuring the new user:

- **Account status:** The user can be enabled or disabled.
- **Username:** defines the username. It cannot start with a number and cannot have special characters except the underscore.
- **User Password:** defines the password for the user, between 8 and 20 characters. Please note that the password cannot be a very simple sequence of characters and must be accepted by the rules defined as per Linux pam_cracklib, which checks the password against dictionary words.
- **Privilege:** defines the privilege level of the user. There are four options available:
    - Administrator
    - Operator
    - ReadOnly
    - NoAccess

In order to add a new user, the system administrator must configure the settings mentioned above and then click on the “Add user” button below.

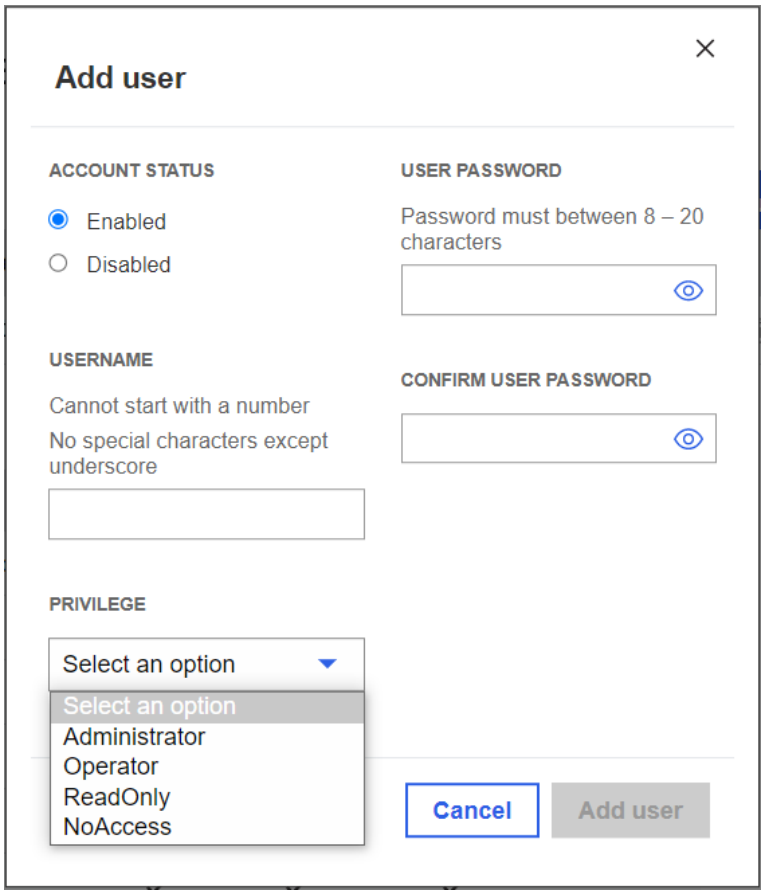


Figure 60: Add user menu

It is possible to check the actions allowed for each privilege level in the “Local Users” menu main page, by clicking on the “Hide privilege role descriptions” button, as shown in the Figure 61.

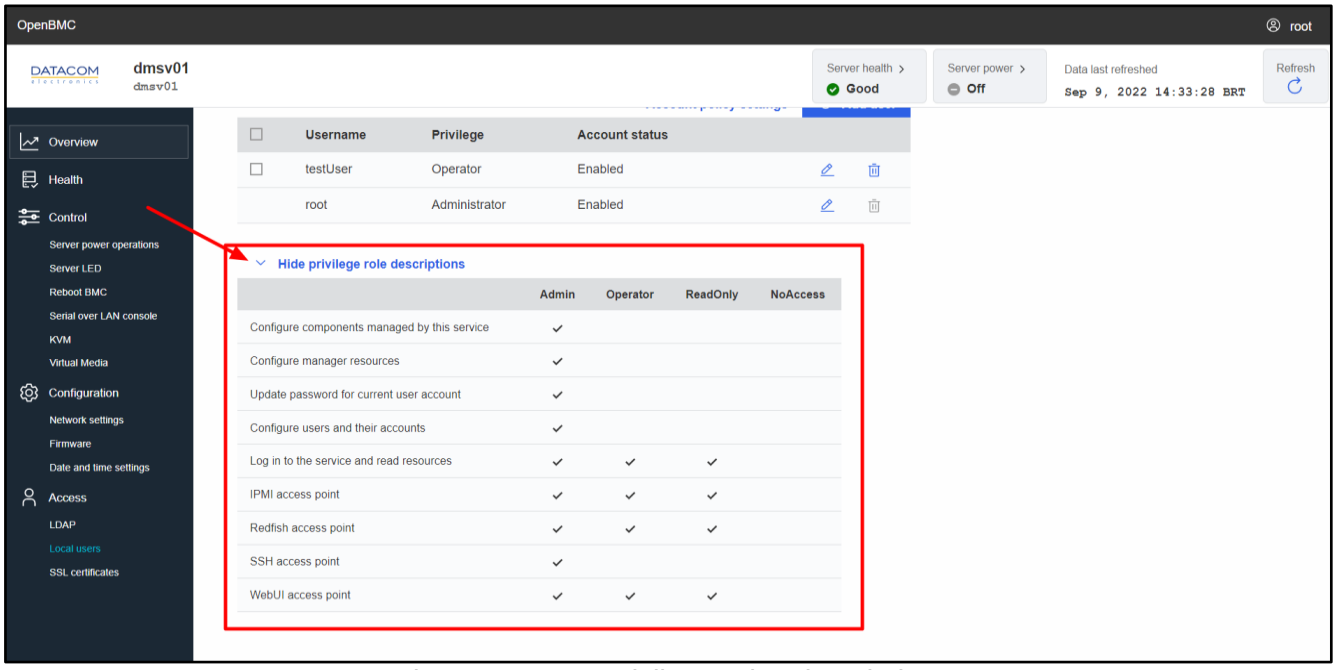


Figure 61: User privilege roles description

After the user has been successfully created, it is possible to edit its settings whenever necessary, by accessing the “Local users” main menu and clicking on the “pencil” symbol related to the user account you would like to edit, as shown in the Figure 62.

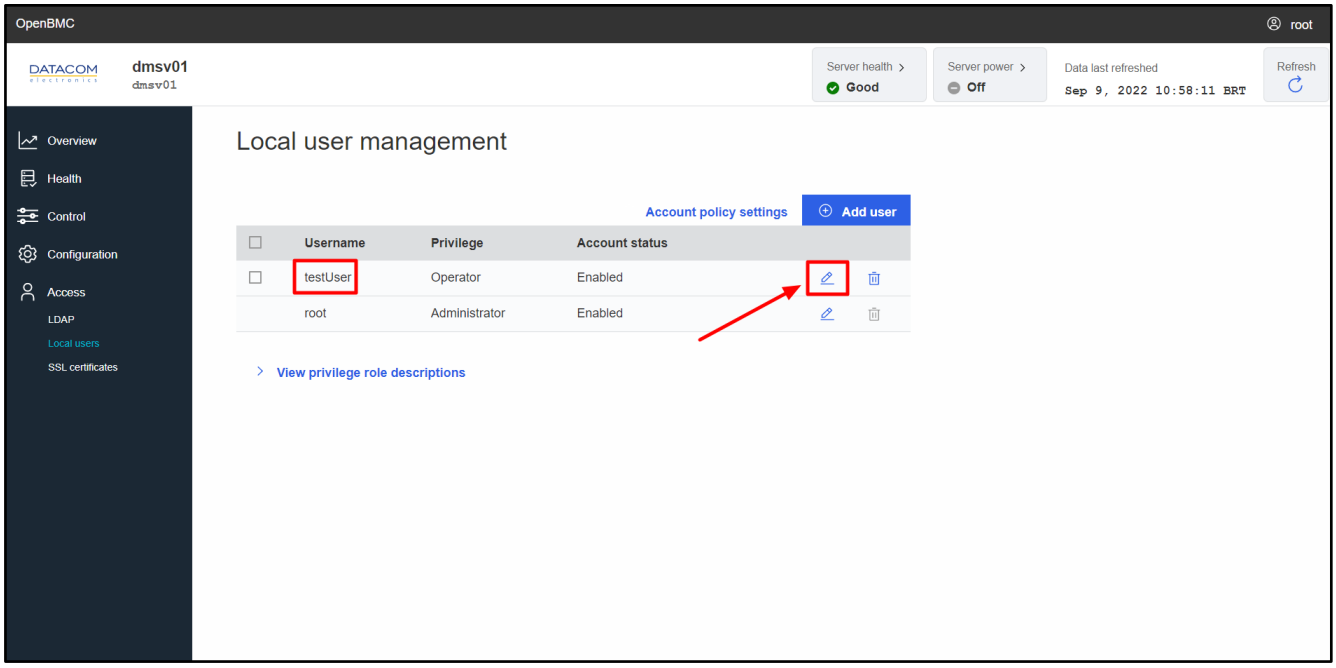


Figure 62: Edit User

In order to delete an user, just click on the “trash” symbol related to the user you would like to delete, as shown in the Figure 63.

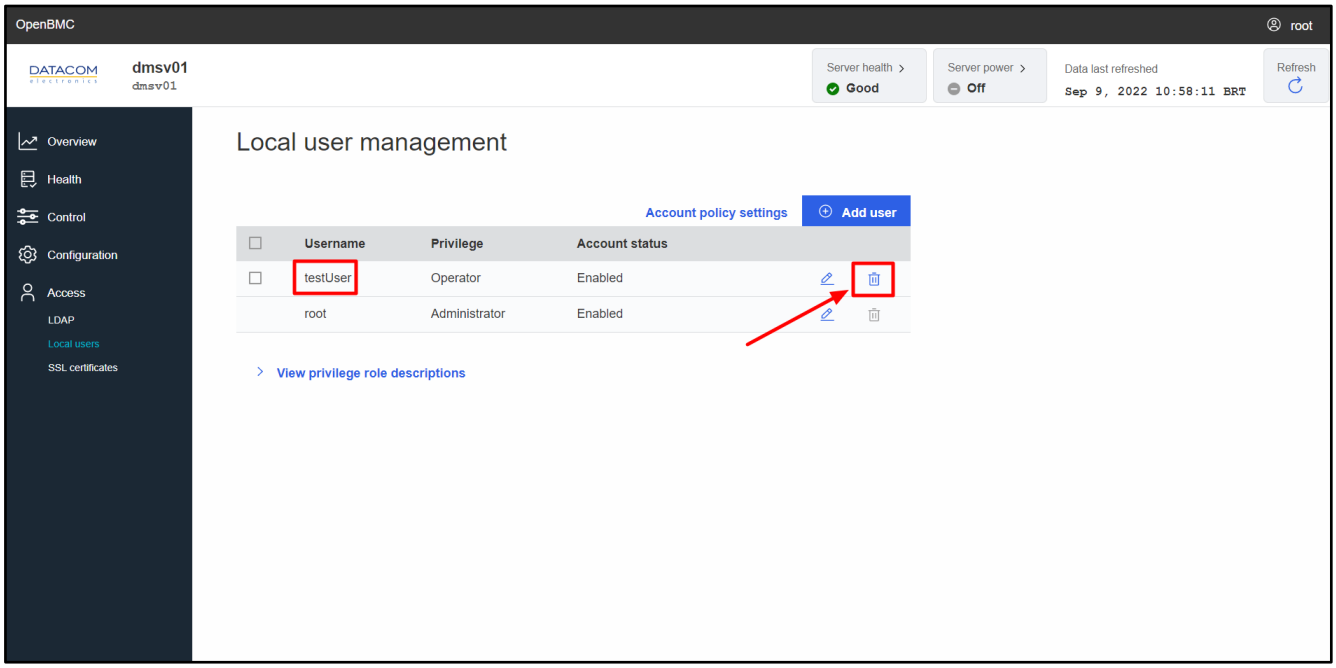


Figure 63: Delete User

Important: the “root” user is a special type of user that cannot be disabled or deleted, cannot have its username modified and the privilege level is always fixed as “Administrator”. The “root” user has always full access to the BMC settings and can only have its password modified.

Another option available in the “Local users” menu is the multiple selection edition option. If the user desires to remove, enable or disable multiple users at the same time, it is possible to select them by means of the particular checkboxes on the left side of the username or select all the users at once by clicking on the checkbox in the header. After selecting the users, it is possible to use the buttons on the header to enable, disable or delete the users, as well as use the “Cancel” button to undo the multiple selection. The “root” user cannot be selected using this feature, because it cannot be enabled, disabled or deleted.

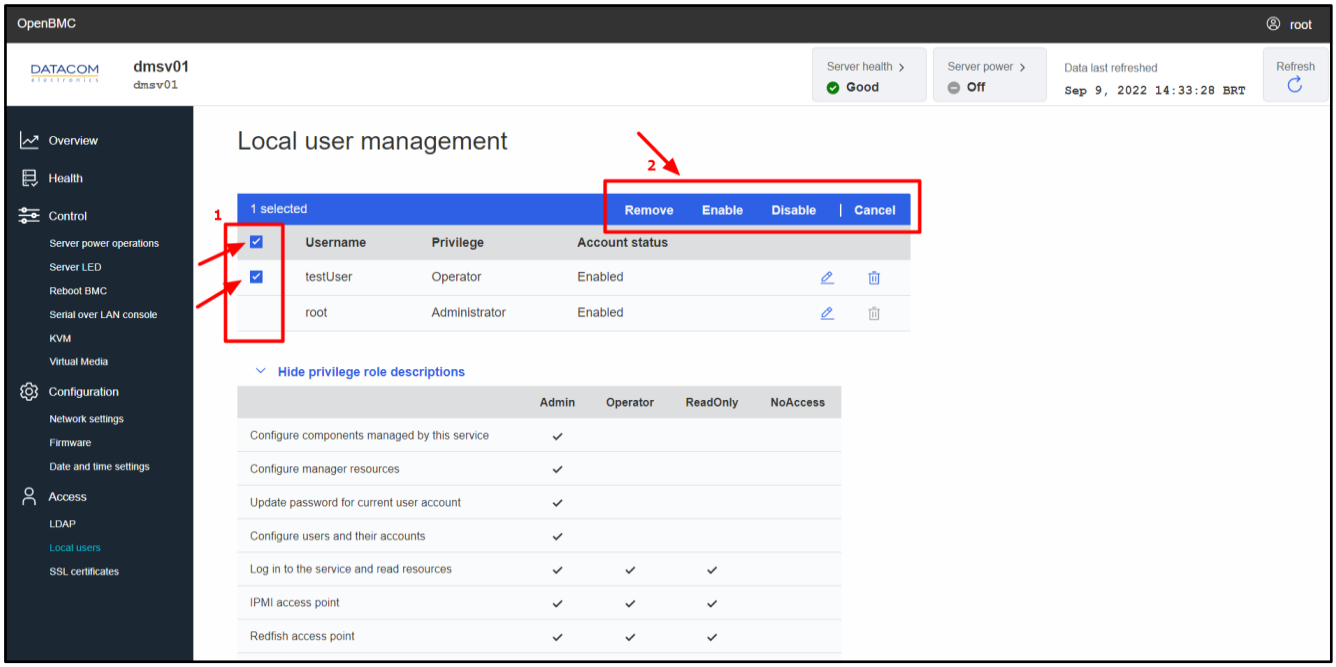


Figure 64: Editing multiple users

2.5.3 SSL certificates

The “SSL certificates” menu allows the user to add certificates or to generate a new CSR (Certificate Signing Request).

2.5.3.1 Adding or replacing a certificate

In order to add a new certificate, the user must click on the “Add new certificate” button, as shown in the Figure 65.

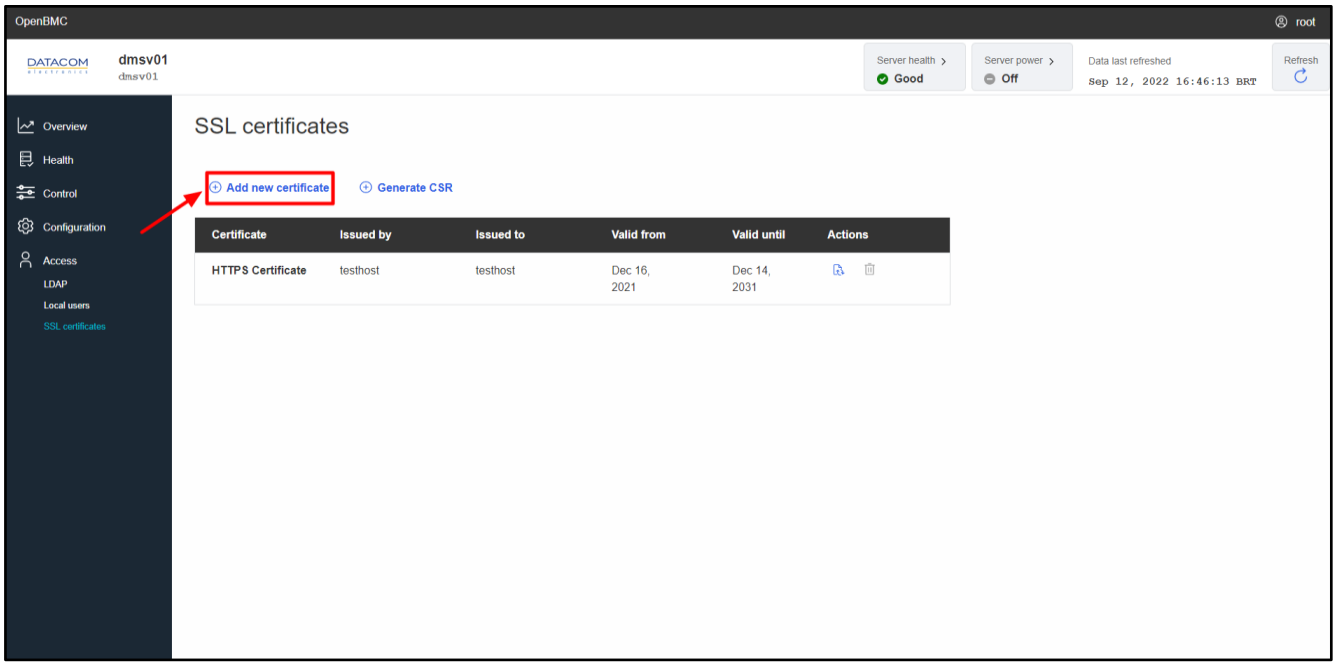


Figure 65: SSL - Add certificate button

By clicking in the “Add new certificate” button, the window shown in the Figure 66 is displayed. The user must select the certificate type (LDAP or CA) and then click on “Choose file” in order to browse the certificate file in the workstation. Then, the certificate addition can be confirmed by clicking on the “Save” button below.

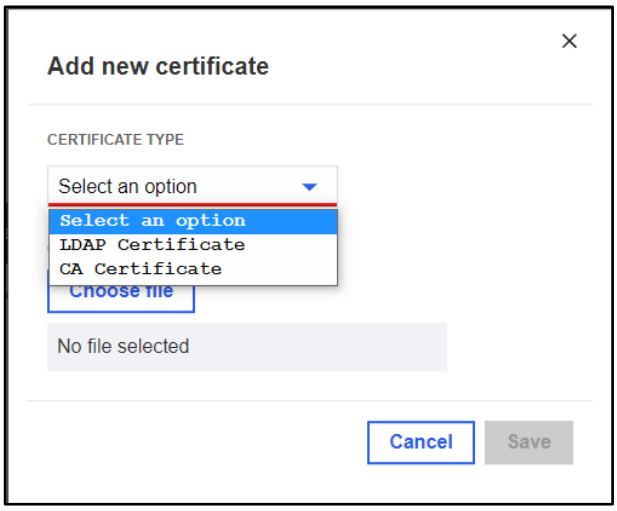


Figure 66: SSL - Add certificate window

The user can also replace an already loaded certificate. This can be done by clicking on the replace icon shown in the Figure 67, then clicking on the “Choose file” button to browse the new certificate file and finally confirming the operation by clicking on the “Replace” button.

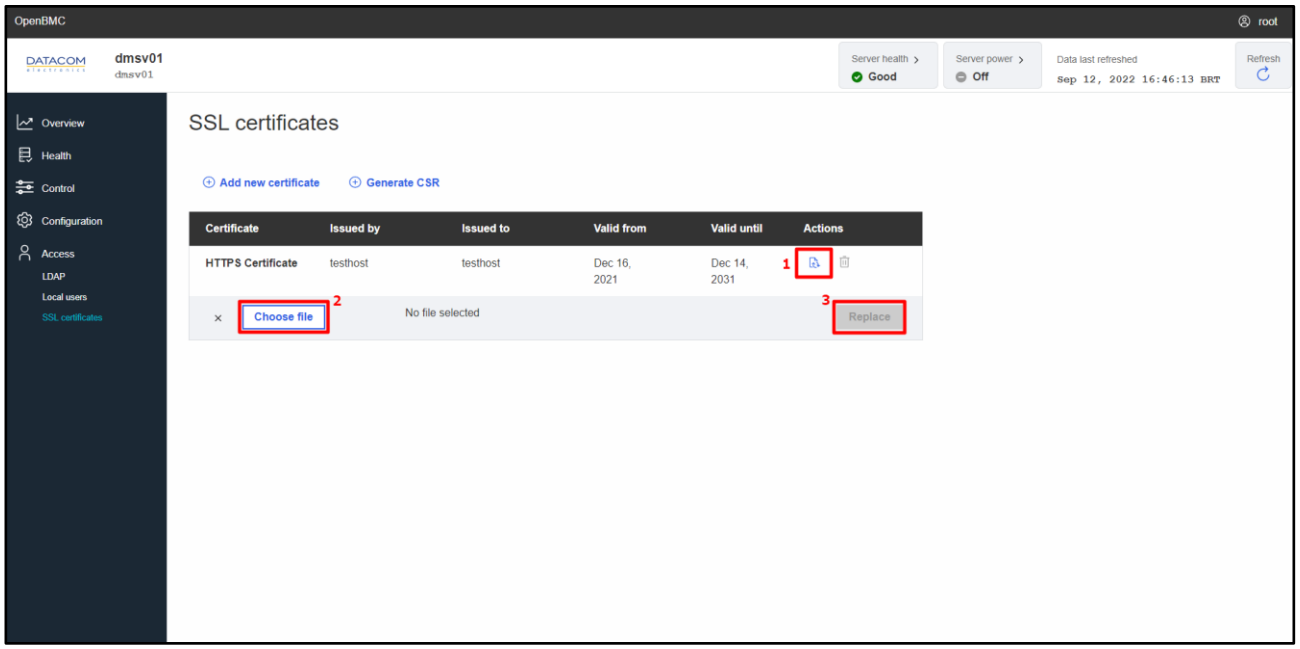


Figure 67: SSL - Replacing the certificate

2.5.3.2 CSR Generation

By clicking on the “Generate CSR” button, as shown in the Figure 68, the user is able to generate a Certificate Signing Request.

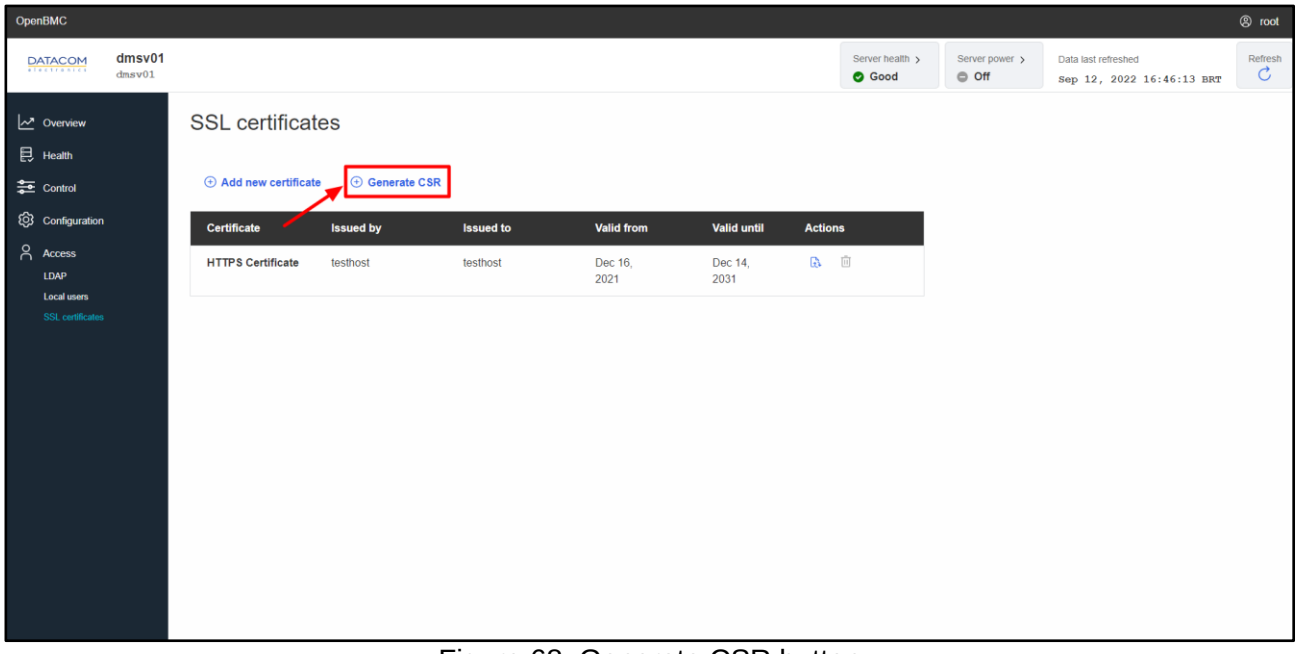


Figure 68: Generate CSR button

When clicking on the “Generate CSR” button, the window shown in the Figure 69 is displayed. The user can select between two types of certificate: HTTPS certificate or LDAP certificate. The LDAP certificate can be used to configure the secure LDAP, as described in the section “2.5.1.1 Enabling and configuring the LDAP”.

Figure 69: CSR generation window

There are also two types of private key algorithms for selection: EC or RSA, available for selection in the right corner of the window.

Figure 70: CSR - Key algorithm selection

Depending on the selection, an additional field is shown for configuration:

- If EC is set, the user is prompted to select the “Key Curve ID”. The available values are “None”, “prime256v1”, “secp521r1” and “secp384r1”.
- If RSA is set, the user is prompted to select the “Key Bit Length”. The only option available in this case is “2048”.

Figure 71: CSR - EC key curve ID configuration

Figure 72: CSR - RSA key bit length configuration

After selecting the certificate type and private key algorithm, the user must fill out the form manually and then click on “Generate CSR” to confirm the operation.

3 Redfish API

The Redfish API is available at the DM-SV01 BMC as a standard way to manage several system resources using the HTTP. The DM-SV01 redfish schema can be accessed by a web browser by means of the URL below:

- https://<BMC_IP>/redfish/v1

After accessing the link above, the user is able to navigate throughout the links of the DM-SV01 redfish interface in the web browser and access the resources available at the BMC, such as sensors, system inventory data, etc.

The sections “3.1 HTTP Methods” and “3.2 HTTP responses” show general information regarding HTTP methods and responses used for accessing the redfish resources of the BMC. The section “3.3 Using Redfish with RESTful APIs” lists several available management resources and how to interact with each one of them.

3.1 HTTP Methods

The HTTP methods below are available for the user to implement specific actions using the Redfish interface.

Method	Description
POST	Create a resource in the specified resource collection.
GET	Retrieve a resource.
PUT	Replace an existing resource.
PATCH	Partially modify or update an existing resource.
DELETE	Remove completely a resource.

Table 4: HTTP resources description

3.2 HTTP responses

When an HTTP method is issued by the client, the server (BMC) answers the request with a corresponding status code, depending on the result of the operation. The response status codes are divided into five groups, each one starting with a specific digit.

- 1xx: Informational responses: the request was received, but the process is still continuing.
- 2xx: Successful responses: the request was received, understood and accepted.
- 3xx: Redirection messages: further actions are required to complete the request.
- 4xx: Client error responses: the request contains invalid syntax or cannot be fulfilled.
- 5xx: Server error responses: the request is aparentelly valid but cannot be fulfilled.

The tables from section “3.2.1 Informational Status Codes” list all response codes available. The response codes are defined by the RFC9110 (3).

3.2.1 Informational Status Codes

Code	Description
100	Continue
101	Switching Protocols
102	Processing
103	Early Hints
104-199	Unassigned

Table 5: HTTP Informational status codes

3.2.2 Successful Status Codes

Code	Description
200	OK
201	Created
202	Accepted
203	Non-Authoritative Information
204	No Content
205	Reset Content
206	Partial Content
207	Multi-Status
208	Already Reported
209-225	Unassigned
226	IM Used
227-299	Unassigned

Table 6: HTTP Successful status codes

3.2.3 Redirection Status Codes

Code	Description
300	Multiple Choices
301	Moved Permanently
302	Found
303	See Other
304	Not Modified
305	Use Proxy

306	(Unused)
307	Temporary Redirect
308	Permanent Redirect
309-399	Unassigned

Table 7: HTTP Redirection status codes

3.2.4 Client Error Status Codes

Code	Description
400	Bad Request
401	Unauthorized
402	Payment Required
403	Forbidden
404	Not Found
405	Method Not Allowed
406	Not Acceptable
407	Proxy Authentication Required
408	Request Timeout
409	Conflict
410	Gone
411	Length Required
412	Precondition Failed
413	Content Too Large
414	URI Too Long
415	Unsupported Media Type
416	Range Not Satisfiable
417	Expectation Failed
418	(Unused)
419-420	Unassigned
421	Misdirected Request
422	Unprocessable Content
423	Locked
424	Failed Dependency
425	Too Early
426	Upgrade Required
427	Unassigned
428	Precondition Required

429	Too Many Requests
430	Unassigned
431	Request Header Fields Too Large
432-450	Unassigned
451	Unavailable For Legal Reasons
452-499	Unassigned

Table 8: HTTP Client Error status codes

3.2.5 Server Error Status Codes

Code	Description
500	Internal Server Error
501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Gateway Timeout
505	HTTP Version Not Supported
506	Variant Also Negotiates
507	Insufficient Storage
508	Loop Detected
509	Unassigned
510	Not Extended (OBSOLETED)
511	Network Authentication Required
512-599	Unassigned

Table 9: HTTP Server Error status codes

3.3 Using Redfish with RESTful APIs

The user can send commands to redfish by using a standard RESTful API of your choice. The examples shown in this document are using Postman, but the procedure is similar when using other APIs.

The table below lists all the requests available in the BMC. Every request has a specific section explaining its functionality and an example of how to perform the operation.

Function	Method	Section
BMC Login	POST	3.3.1 Session Login
BMC Logout	DELETE	3.3.3 Session Logout

Inventory - Mainboard	GET	3.3.4.1 Mainboard Inventory
Inventory - CPUs	GET	3.3.4.2 Processors Inventory
Detailed Inventory - CPUs	GET	3.3.4.3 Detailed inventory about a specific processor
Inventory - DDR memories	GET	3.3.4.4 Memory modules inventory
Detailed Inventory - DDR memories	GET	3.3.4.5 Detailed inventory information about specific memory module
Inventory - storage devices	GET	3.3.4.6 Storage inventory
Detailed Inventory - storage devices	GET	3.3.4.7 Detailed inventory about a specific storage device
Power sensors	GET	3.3.5.1 Power Sensors
Temperature sensors	GET	3.3.5.2 Temperature Sensors
System Total Current Consumption	GET	3.3.5.3.1 Current Consumption
System Total Power Consumption	GET	3.3.5.3.2 Power Consumption
System Peak Power	GET	3.3.5.3.3 Peak Power
Reset Peak Power sensor	POST	3.3.5.4 Peak Power sensor reset
Reset Energy sensor	POST	3.3.5.5 Energy sensor reset
Turn on Server Indicator LED	PATCH	3.3.6.1 Turn on Indicator LED
Turn off Server Indicator LED	PATCH	3.3.6.2 Turn off Indicator LED
Host power on	POST	3.3.7.1 Power On Host
Host power off	POST	3.3.7.2 Power Off Host
Host restart	POST	3.3.7.3 Restart Host
Host forced power off	POST	3.3.7.4 Force Power Off Host
Host forced restart	POST	3.3.7.5 Force Restart Host
Network settings	PATCH	3.3.8 Network Settings
Boot override - PXE	PATCH	3.3.9.1 Force PXE Boot Override
Boot override - CD	PATCH	3.3.9.2 Force CD-ROM/Virtual Media Boot Override
Boot override - BIOS/UEFI setup	PATCH	3.3.9.3 Force BIOS Setup Boot Override
Boot override - USB	PATCH	3.3.9.4 Force USB Boot Override
Boot override - HDD	PATCH	3.3.9.5 Force HDD Boot Override
Boot override - Disable	PATCH	3.3.9.6 Disable Boot Override
Configure Open LDAP	PATCH	3.3.10.1 Open LDAP
Configure LDAP - Active Directory	PATCH	3.3.10.2 Active Directory
Configure LDAP - Role Groups	PATCH	3.3.10.3 Role Groups
Change root password	PATCH	3.3.11.1 Change root password

Add new BMC user	POST	3.3.11.2 Add BMC User
Change BMC user role	PATCH	3.3.11.3 Change BMC User Role
Change BMC user password	PATCH	3.3.11.4 Change BMC User Password
Delete BMC user	DELETE	3.3.11.5 Delete BMC User
Update BMC FW	POST	3.3.12.1 Update BMC Firmware
Update BIOS FW	POST	3.3.12.2 Update BIOS Firmware
View log entries	GET	3.3.13.1 View Log Entries
Clear log entries	POST	3.3.13.2 Delete Log Entries
BMC reboot	POST	3.3.14.1 Reboot BMC
BMC factory reset	POST	3.3.14.2 Reset BMC to Factory Defaults

3.3.1 Session Login

The user can perform a session login authentication in the BMC by means of the POST operation described below. The username and password must be sent inside the payload field.

Function	BMC Login
Operation	POST
URI	https://<BMC_IP>/redfish/v1/SessionService/Sessions

Payload	{
	“UserName” : “<username>”,
	“Password” : ”<password>”
	}

Header	None
Expected response	201 created

Reply	{
	"X-Auth-Token": "<token>",
	“Location”: “/redfish/v1/SessionService/Sessions/<location_id>”
	}

Once the operation is successful, it returns the response “201 Created”. The session location is then created with a corresponding ID and an X-Auth-Token is generated.

Important: The X-Auth-Token is used in the header of further redfish requests as an authorization ID.

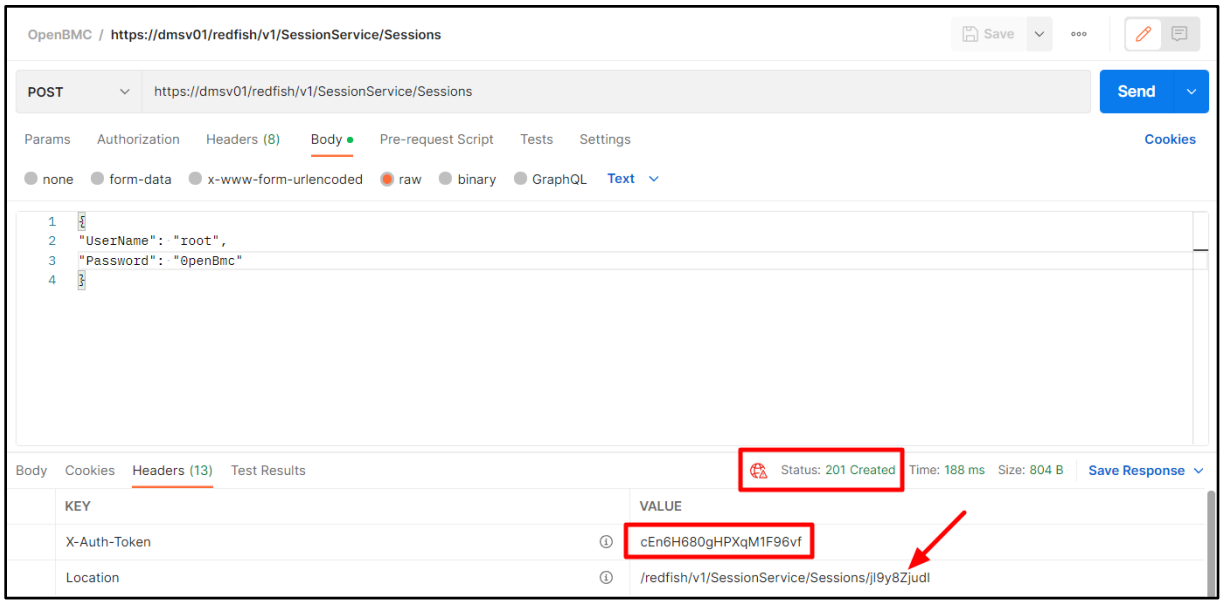


Figure 73: Redfish session create - POST

3.3.2 Using the X-Auth-Token

The X-Auth-Token generated by following the procedure described in section 3.3.1 Session Login can be used to authenticate every request sent to the BMC by means of the Redfish interface. As an example, consider the token generated by the session login shown in the Figure 74. When using Postman, the token is available in the “Headers” tab of the request response workspace.

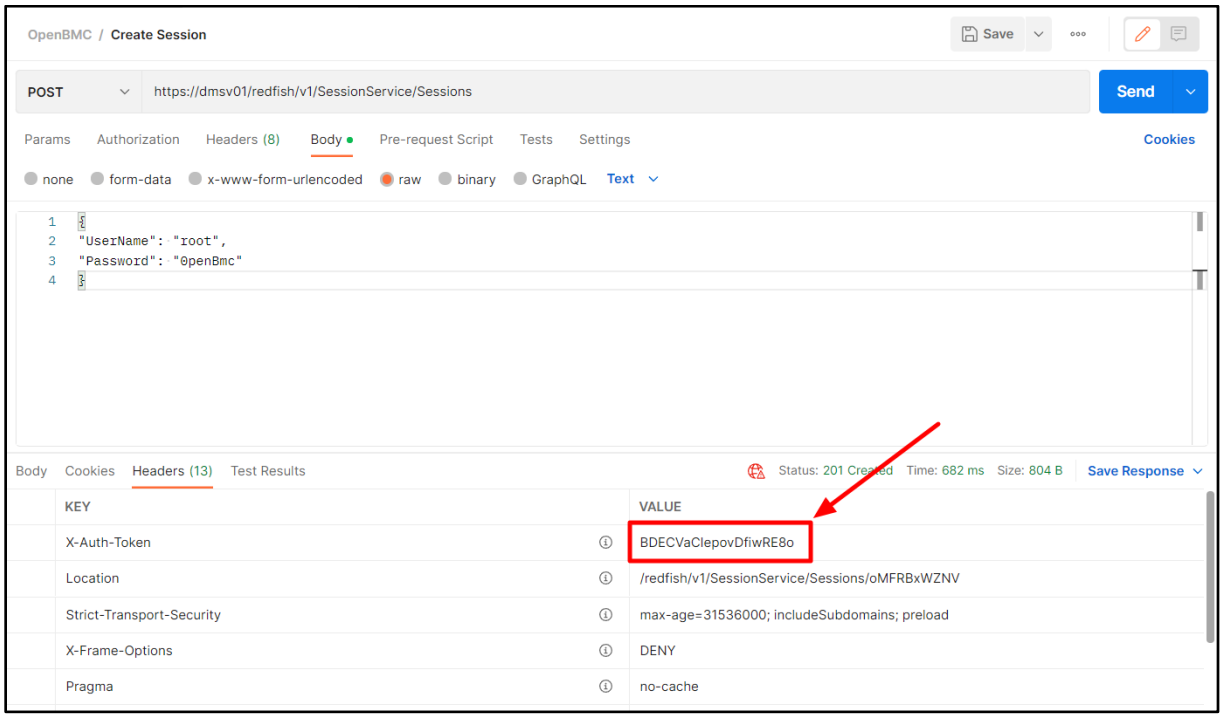


Figure 74: BMC session authentication token

When performing further requests, this token must be added to the header for authentication. As an example, this can be done in Postman by means of the “Headers" tab in the request parameters workspace, as shown in Figure 75.

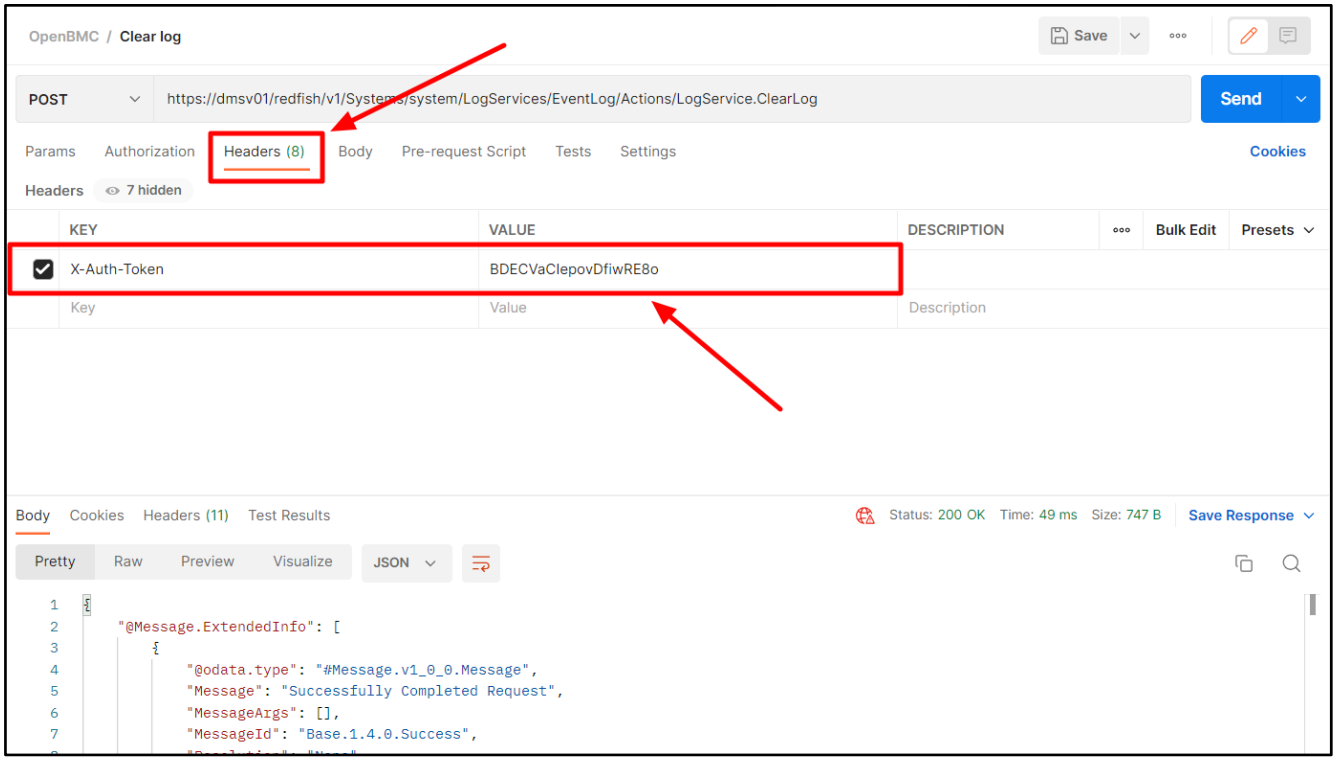


Figure 75: Adding the token for authentication

3.3.3 Session Logout

The user can perform a session logout in the BMC by means of the DELETE operation described below. In the URI, it is required to add the session ID to be deleted. As an example, the session created in section “3.3.1 Session Login” could be deleted by using the URI below:

- *https://<BMC_IP>/redfish/v1/SessionService/Sessions/jl9y8Zjudl.*

Function	BMC Logout
Operation	DELETE
URI	https://<BMC_IP>/redfish/v1/SessionService/Sessions/<session_ID>
Payload	None
Header	X-Auth-Token: “<token>”
Expected response	200 OK

Reply	{
	"@odata.id": "/redfish/v1/SessionService/Sessions/<location>",
	"@odata.type": "#Session.v1_0_2.Session",
	"Description": "Manager User Session",
	"Id": "<location>",
	"Name": "User Session",
	"UserName": "<user>"
	}

Once the operation is successful, it returns the response “200 OK” and the corresponding session is removed from the BMC.

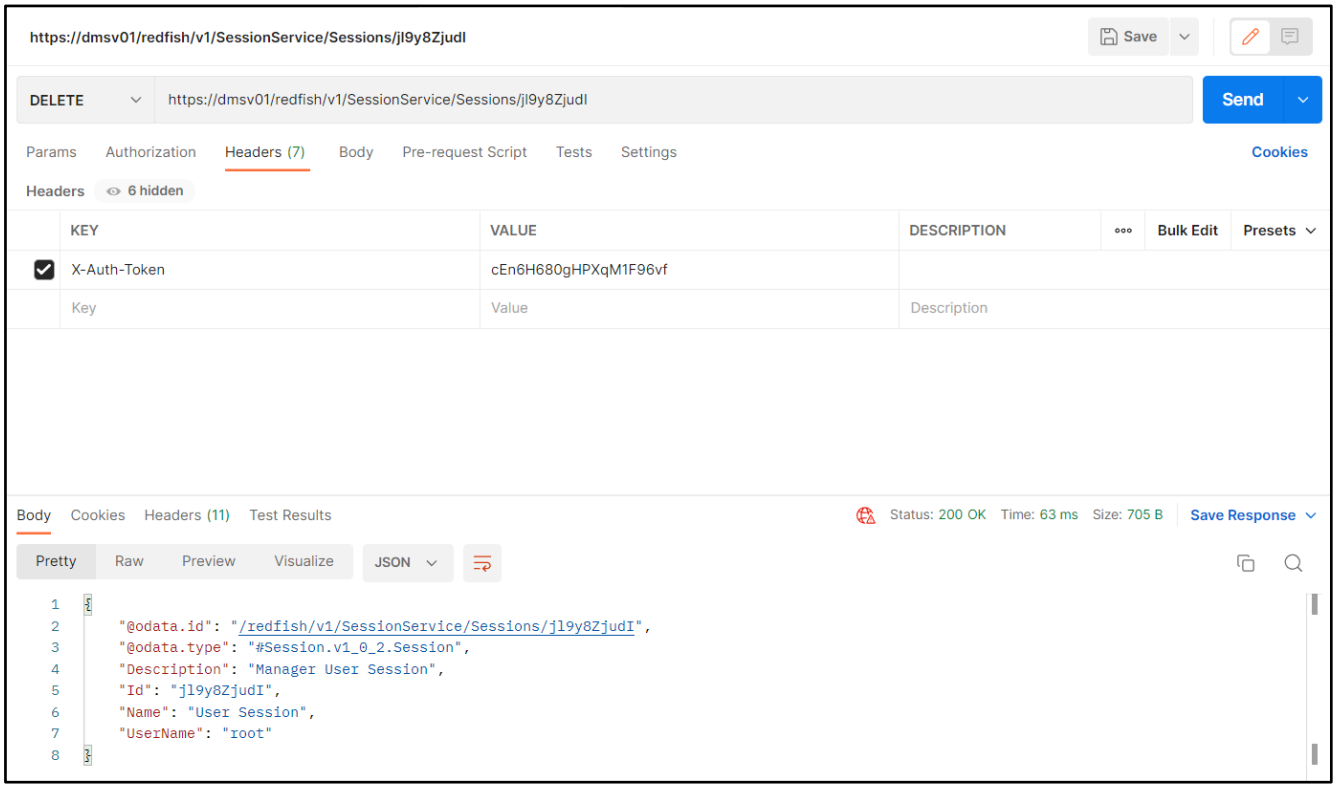


Figure 76: Redfish session remove - DELETE

3.3.4 System Inventory

The “System Inventory” section describes the resources used to retrieve information about several system components, such as motherboard, CPUs, memories, storage devices, etc.

3.3.4.1 Mainboard Inventory

Using a GET request, it is possible to retrieve inventory information from the DM-SV01 mainboard.

Function	Inventory - Mainboard
Operation	GET
URI	https://<BMC_IP>/redfish/v1/Chassis/motherboard
Payload	none
Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Please, see the example below.

As an example, the excerpt below shows the data provided by the DM-SV01 motherboard using the redfish GET request. The user can check relevant information such as indicator LED status, part number, serial number, power state, system health status, etc.


```
{
   "@odata.id": "/redfish/v1/Chassis/motherboard",
   "@odata.type": "#Chassis.v1_10_0.Chassis",
   "ChassisType": "RackMount",
   "Id": "motherboard",
   "IndicatorLED": "Off",
   "Links": {
      "ComputerSystems": [
         {
            "@odata.id": "/redfish/v1/Systems/system"
         }
      ],
      "ManagedBy": [
         {
            "@odata.id": "/redfish/v1/Managers/bmc"
         }
      ]
   },
   "Manufacturer": "Datacom",
   "Model": "DM-SV01 - Mainboard",
   "Name": "motherboard",
   "PCIeDevices": {
      "@odata.id": "/redfish/v1/Systems/system/PCIeDevices"
   },
   "PartNumber": "750.0525.61",
   "Power": {
      "@odata.id": "/redfish/v1/Chassis/motherboard/Power"
```

```
    },
    "PowerState": "Off",
    "Sensors": {
        "@odata.id": "/redfish/v1/Chassis/motherboard/Sensors"
    },
    "SerialNumber": "5676848",
    "Status": {
        "Health": "OK",
        "HealthRollup": "OK",
        "State": "StandbyOffline"
    },
    "Thermal": {
        "@odata.id": "/redfish/v1/Chassis/motherboard/Thermal"
    }
}
```


Once the operation is successful, it returns the response “200 OK” and the inventory information is retrieved.

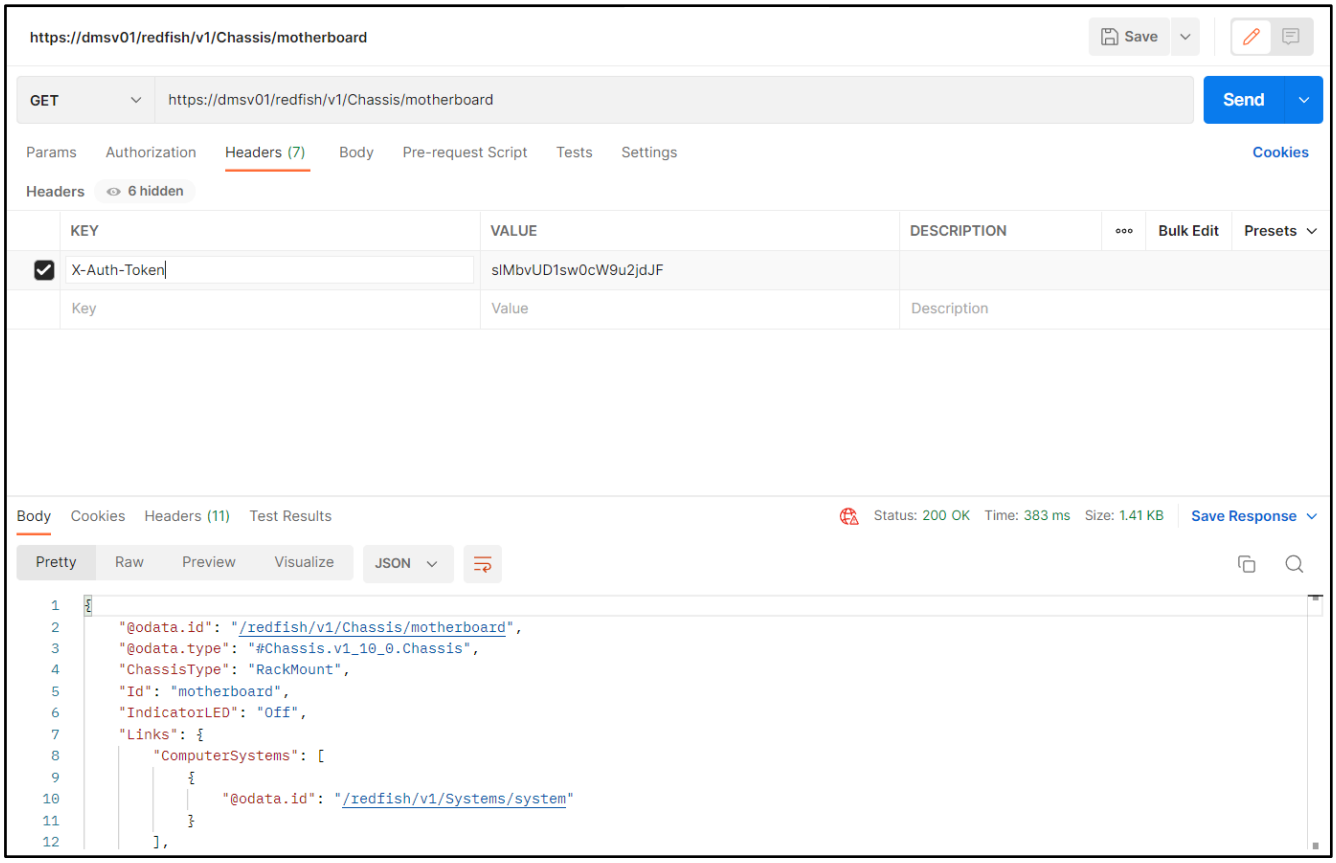


Figure 77: Redfish - Motherboard inventory

3.3.4.2 Processors Inventory

Using a GET request, it is possible to retrieve inventory information from the DM-SV01 CPUs.

Function	Inventory - CPUs
Operation	GET
URI	https://<BMC_IP>/redfish/v1/Systems/system/Processors
Payload	none
Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Please, see the example below.

As an example, the excerpt below shows the data provided by the DM-SV01 mainboard using the redfish GET request, where the user can check the CPU count. Additionally, the user can view the list of “Members”, which are the available CPUs in the system that can be accessed to retrieve more detailed CPU information by using the procedure shown in section “3.3.4.3 Detailed inventory about a specific processor”.


```
{
   "@odata.id": "/redfish/v1/Systems/system/Processors/",
   "@odata.type": "#ProcessorCollection.ProcessorCollection",
   "Members": [
      {
         "@odata.id": "/redfish/v1/Systems/system/Processors/cpu0"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Processors/cpu1"
      }
   ],
   "Members@odata.count": 2,
   "Name": "Processor Collection"
}
```


Once the operation is successful, it returns the response “200 OK” and the inventory information is retrieved.

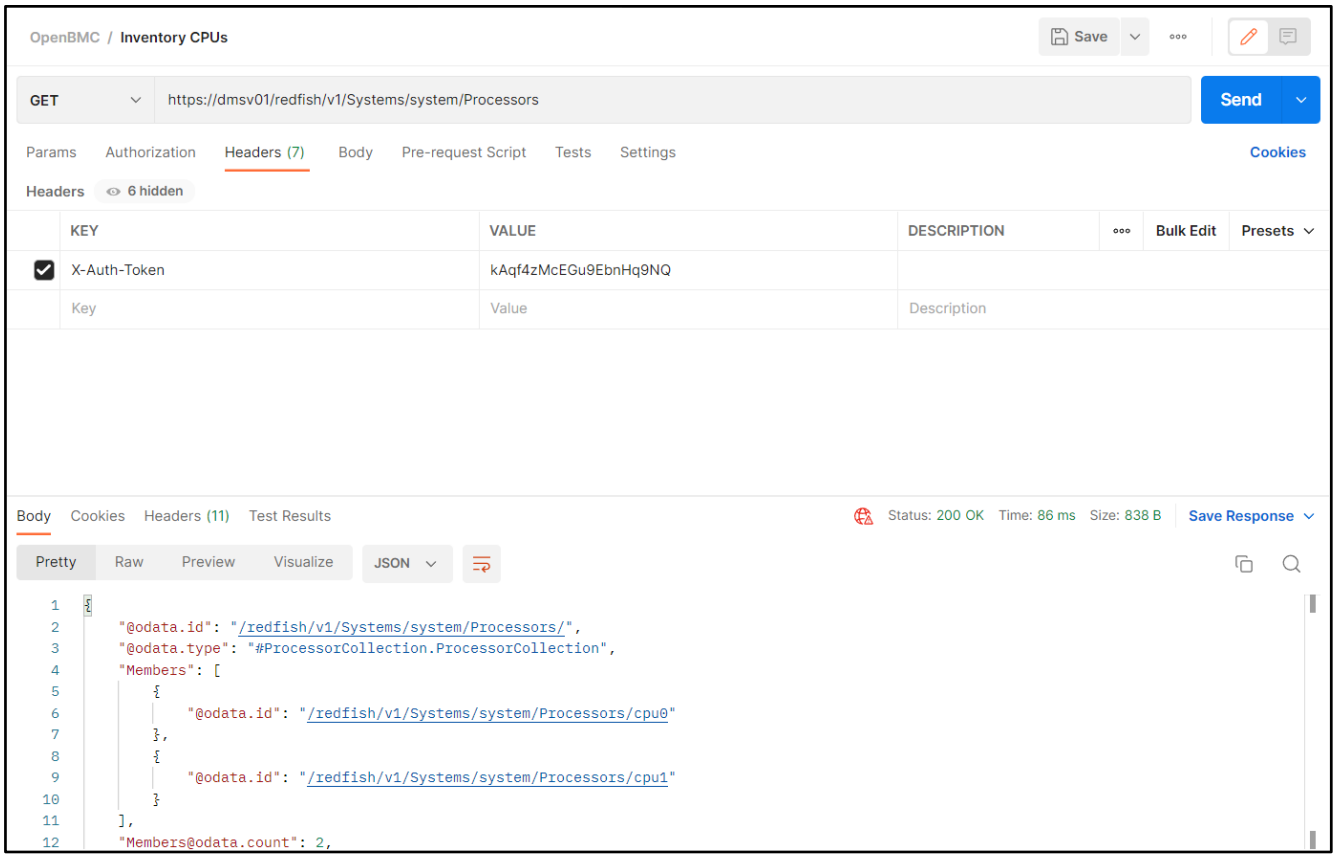


Figure 78: Redfish - CPUs inventory

3.3.4.3 Detailed inventory about a specific processor

Using a GET request, it is possible to retrieve detailed inventory information from a specific CPU of the DM-SV01 server. The user can check which CPUs are available for requesting the inventory by using the procedure described in section “3.3.4.2 Processors Inventory”. Anyway, the CPUs available by default are “cpu0” and “cpu1” in a two socket system.

Function	Detailed Inventory - CPUs
Operation	GET
URI	https://<BMC_IP>/redfish/v1/Systems/system/Processors/<cpu>
Payload	none
Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Please, see the example below.

As an example, the excerpt below shows the data provided by the redfish when using the GET request to retrieve information about the CPU named as “cpu0”. The redfish provides detailed information about the CPU, such as manufacturer, model, frequency, serial number, number of cores and threads, etc.

```
{
   "@odata.id": "/redfish/v1/Systems/system/Processors/cpu0",
   "@odata.type": "#Processor.v1_7_0.Processor",
   "Id": "cpu0",
   "InstructionSet": "x86-64",
   "Manufacturer": "AMD",
   "MaxSpeedMHz": 3200,
   "Model": "EPYC",
   "Name": "AMD EPYC 7282 16-Core Processor                ",
   "PartNumber": "",
   "ProcessorArchitecture": "x86",
   "ProcessorType": "CPU",
   "SerialNumber": "OFF0W2Y",
   "Status": {
      "Health": "OK",
      "State": "Enabled"
   },
   "TotalCores": 16,
   "TotalThreads": 32
}
```


Once the operation is successful, it returns the response “200 OK” and the inventory information is retrieved.

If the CPU is not present in the system, the field "State” is shown as “Absent” and no inventory information is retrieved.

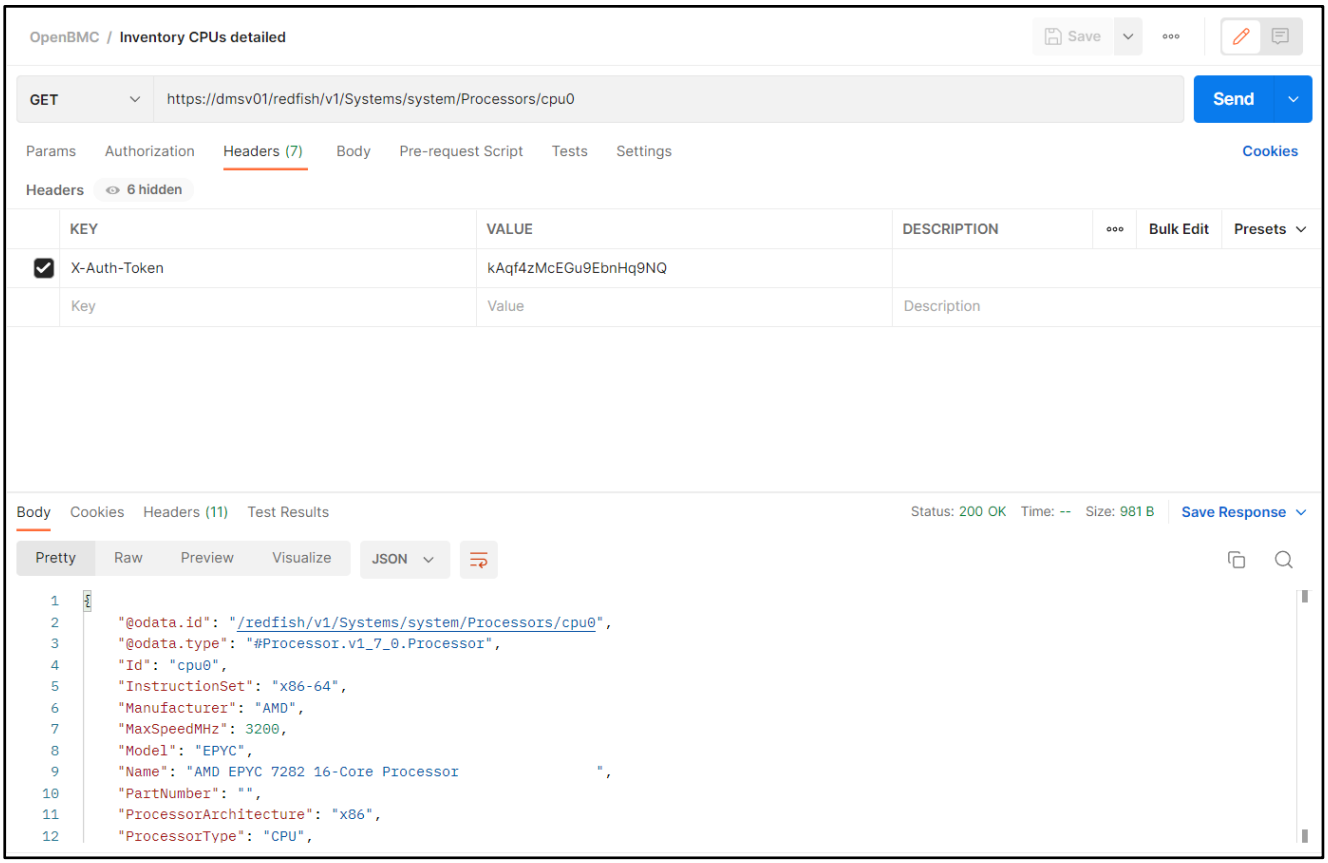


Figure 79: Redfish - Specific CPU detailed Inventory

3.3.4.4 Memory modules inventory

Using a GET request, it is possible to retrieve inventory information from the DM-SV01 DDR memory modules.

Function	Inventory - DDR memories
Operation	GET
URI	https://<BMC_IP>/redfish/v1/Systems/system/Memory
Payload	none
Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Please, see the example below.

As an example, the excerpt below shows the data provided by the DM-SV01 mainboard using the redfish GET request, where the user can check the memory count. Additionally, the user can view the list of “Members”, which are the available memories in the system that can be accessed to retrieve more detailed memory information by using the procedure shown in section “3.3.4.5 Detailed inventory information about specific memory module”.

```
{
   "@odata.id": "/redfish/v1/Systems/system/Memory/",
   "@odata.type": "#MemoryCollection.MemoryCollection",
   "Members": [
      {
         "@odata.id": "/redfish/v1/Systems/system/Memory/dimm0"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Memory/dimm1"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Memory/dimm10"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Memory/dimm11"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Memory/dimm12"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Memory/dimm13"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Memory/dimm14"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Memory/dimm15"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Memory/dimm2"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Memory/dimm3"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Memory/dimm4"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Memory/dimm5"
      },
      {
```

```
            "@odata.id": "/redfish/v1/Systems/system/Memory/dimm6"
        },
        {
            "@odata.id": "/redfish/v1/Systems/system/Memory/dimm7"
        },
        {
            "@odata.id": "/redfish/v1/Systems/system/Memory/dimm8"
        },
        {
            "@odata.id": "/redfish/v1/Systems/system/Memory/dimm9"
        }
    ],
    "Members@odata.count": 16,
    "Name": "Memory Module Collection"
}
```


Once the operation is successful, it returns the response “200 OK” and the inventory information is retrieved.

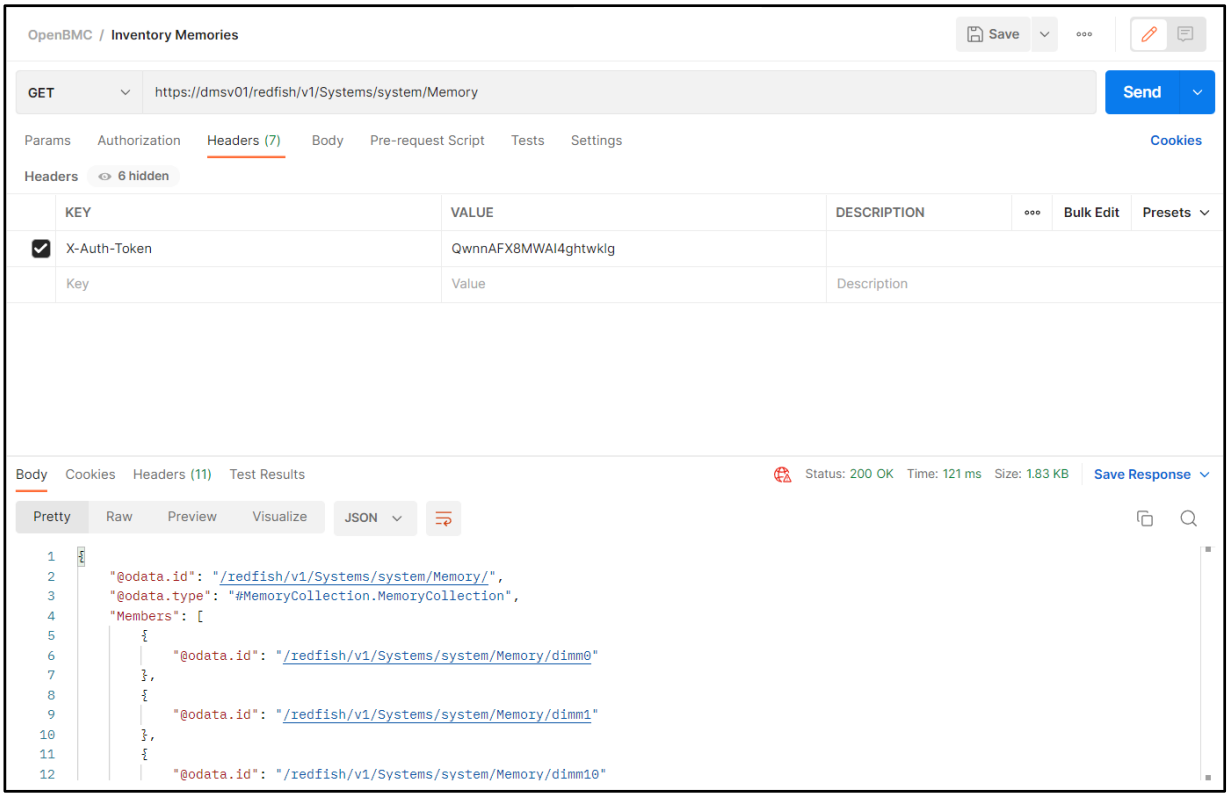


Figure 80: Redfish - Memories inventory

3.3.4.5 Detailed inventory information about specific memory module

Using a GET request, it is possible to retrieve detailed inventory information from a specific DDR memory device installed in the DM-SV01 server. The user can check which memories are available for requesting the inventory by using the procedure described in section “3.3.4.4 Memory modules inventory”. Anyway, the memories available by default are “dimm0” up to “dimm15” in a two socket system.

Function	Detailed Inventory - DDR memories
Operation	GET
URI	https://<BMC_IP>/redfish/v1/Systems/system/Processors/<memory>
Payload	none
Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Please, see the example below.

As an example, the excerpt below shows the data provided by the redfish when using the GET request to retrieve information about the memory named as “dimm3”. The redfish provides detailed information about the memory, such as manufacturer, part number, capacity, serial number, etc.


```
{
   "@odata.id": "/redfish/v1/Systems/system/Memory/dimm3",
   "@odata.type": "#Memory.v1_6_0.Memory",
   "CapacityMiB": 8192,
   "DataWidthBits": 64,
   "Id": "dimm3",
   "Manufacturer": "Hynix",
   "MemoryDeviceType": "xyz.openbmc_project.Inventory.Item.Dimm.DeviceType.DDR4",
   "Name": "DIMM Slot",
   "PartNumber": "HMA81GR7AFR8N-VK   ",
   "SerialNumber": "717B9101",
   "Status": {
      "Health": "OK",
      "State": "Enabled"
   }
}
```


Once the operation is successful, it returns the response “200 OK” and the inventory information is retrieved.

If the memory is not present in the system, the field "State" is shown as “Absent” and no inventory information is retrieved.

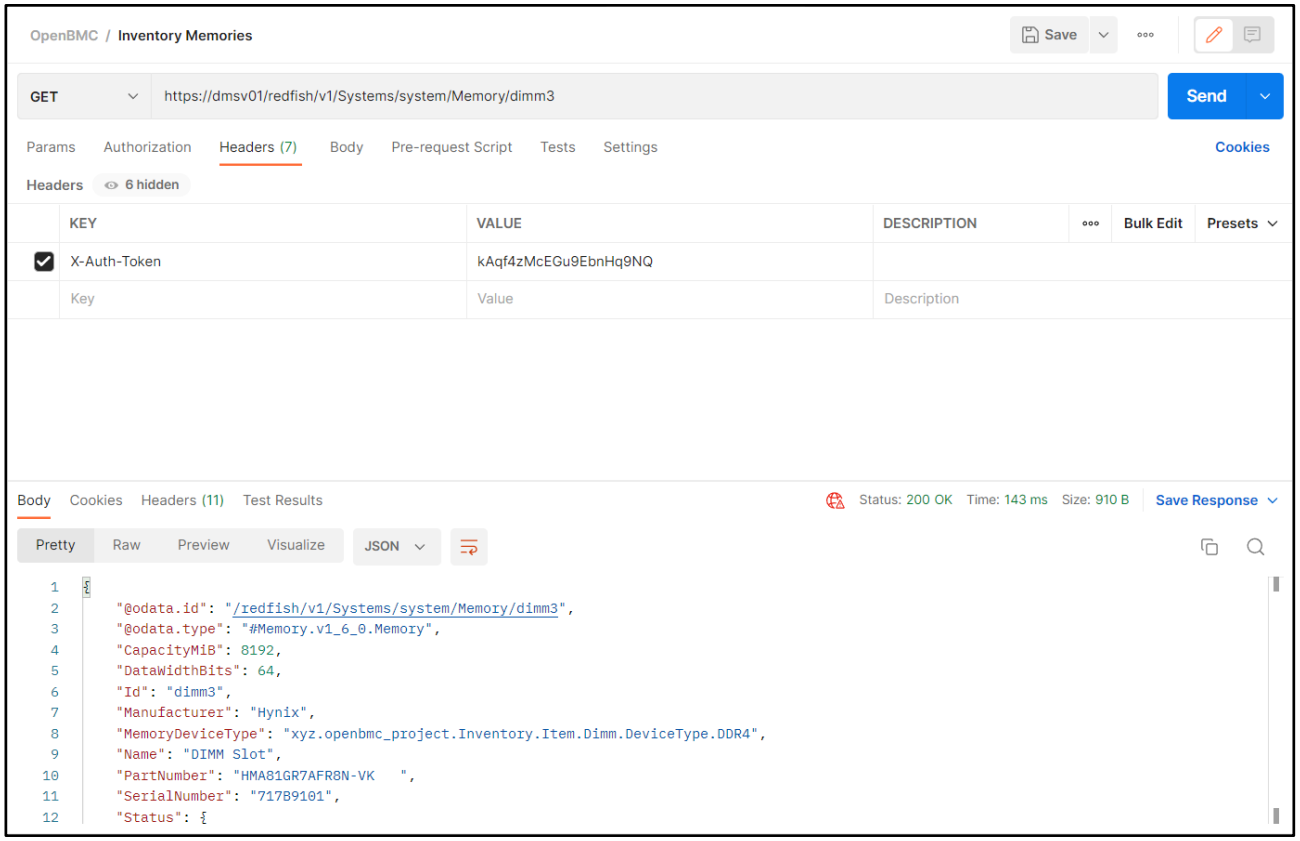


Figure 81: Redfish - Specific memory detailed Inventory

3.3.4.6 Storage inventory

Using a GET request, it is possible to retrieve inventory information from the DM-SV01 storage devices.

Function	Inventory - storage devices
Operation	GET
URI	https://<BMC_IP>/redfish/v1/Systems/system/1/
Payload	none
Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Please, see the example below.

As an example, the excerpt below shows the data provided by the DM-SV01 mainboard using the redfish GET request, where the user can check the storage devices count. Additionally, the user can view the list of “Members”, which are the available storage devices in the system that can be accessed

to retrieve more detailed memory information by using the procedure shown in section “3.3.4.7 Detailed inventory about a specific storage device”.


```
{
   "@odata.id": "/redfish/v1/Systems/system/Storage/1",
   "@odata.type": "#Storage.v1_7_1.Storage",
   "Drives": [
      {
         "@odata.id": "/redfish/v1/Systems/system/Storage/1/Drives/nvme0"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Storage/1/Drives/nvme1"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Storage/1/Drives/nvme10"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Storage/1/Drives/nvme2"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Storage/1/Drives/nvme3"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Storage/1/Drives/nvme4"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Storage/1/Drives/nvme5"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Storage/1/Drives/nvme6"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Storage/1/Drives/nvme7"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Storage/1/Drives/nvme8"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/Storage/1/Drives/nvme9"
      }
   ],
   "Drives@odata.count": 11,
```

```
    "Id": "1",
    "Name": "Storage",
    "Status": {
        "Health": "OK",
        "HealthRollup": "OK",
        "State": "Enabled"
    }
}
```


Once the operation is successful, it returns the response “200 OK” and the inventory information is retrieved.

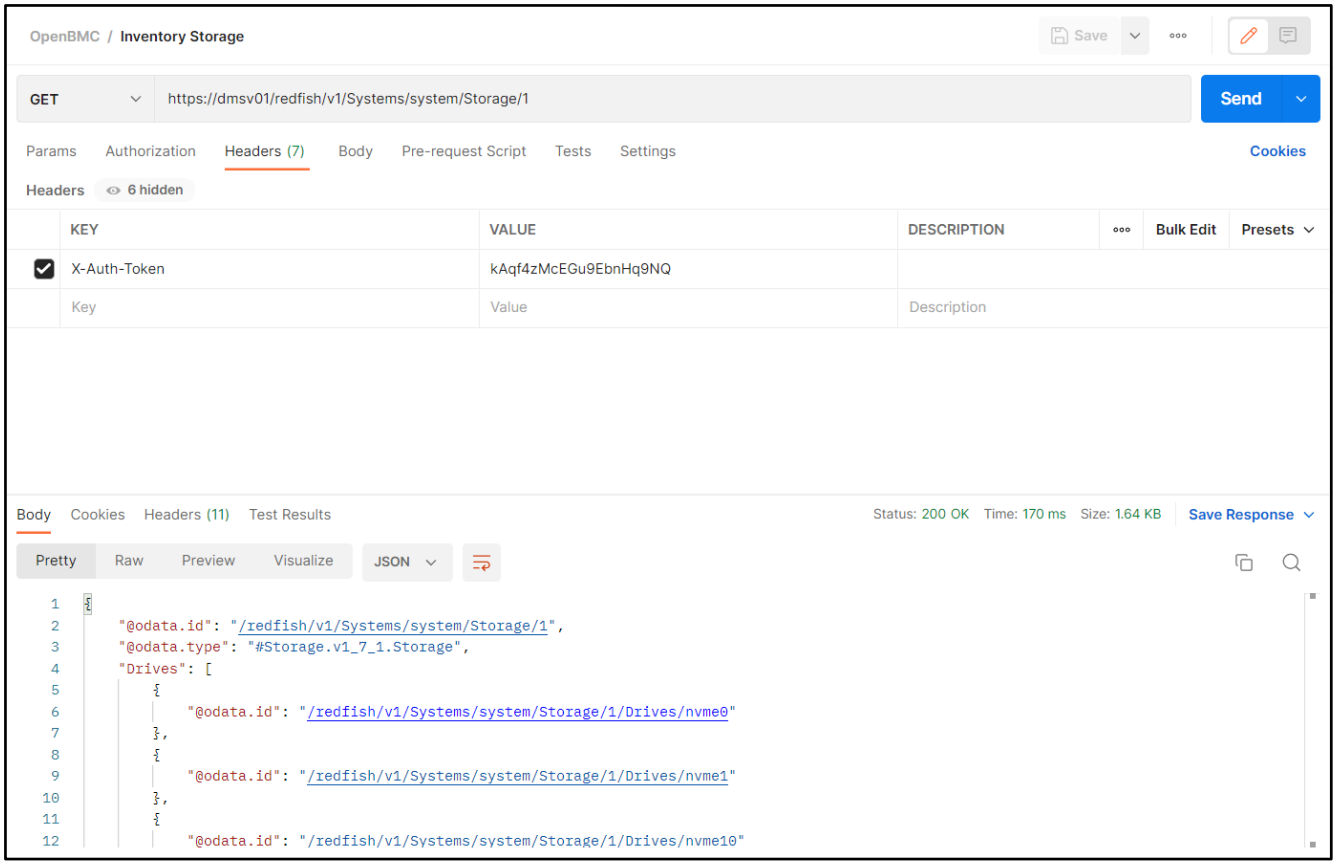


Figure 82: Redfish - Storage devices inventory

3.3.4.7 Detailed inventory about a specific storage device

Using a GET request, it is possible to retrieve detailed inventory information from a specific storage device of the DM-SV01 server. The user can check which storage devices are available for requesting the inventory by using the procedure described in section “3.3.4.6 Storage inventory”. Anyway, the storage devices available by default are “nvme0” up to “nvme10”. Detailed information regarding the storage devices mapping according to their terminology can be found in section “2.2.2 Hardware status”.

Function	Detailed Inventory - storage devices
Operation	GET
URI	https://<BMC_IP>/redfish/v1/Systems/system/Storage/1/Drives/<nvme>
Payload	none
Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Please, see the example below.

As an example, the excerpt below shows the data provided by the redfish when using the GET request to retrieve information about the storage device named as “nvme0” (onboard M.2 nvme disk). The redfish provides detailed information about the storage device, such as manufacturer, model, serial number, etc.


```
{
   "@odata.id": "/redfish/v1/Systems/system/Storage/1/Drives/nvme0",
   "@odata.type": "#Drive.v1_7_0.Drive",
   "Id": "nvme0",
   "Links": {
      "Chassis": {
         "@odata.id": "/redfish/v1/Chassis/chassis"
      }
   },
   "Manufacturer": "VID 126F SSVID 126F",
   "Model": "FLEXXON M.2                ",
   "Name": "nvme0",
   "PartNumber": "",
   "SerialNumber": "F210630OS14600092",
   "Status": {
      "Health": "OK",
      "HealthRollup": "OK",
      "State": "Enabled"
   }
}
```


Once the operation is successful, it returns the response “200 OK” and the inventory information is retrieved.

If the storage device is not present in the system, the field "State” is shown as “Disabled” and no inventory information is retrieved.

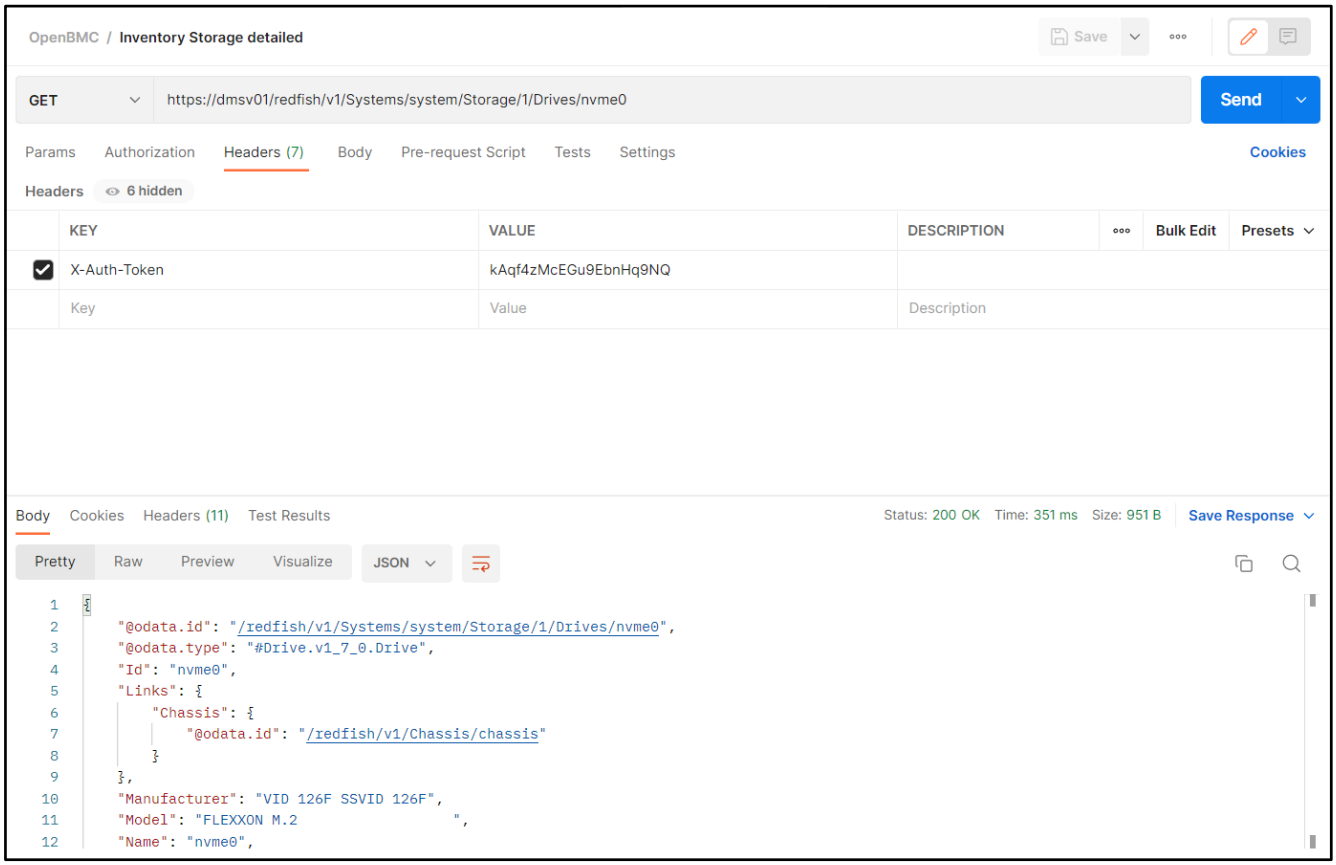


Figure 83: Redfish - Specific storage device detailed Inventory

3.3.5 Sensors

The power and temperature sensors available in the DM-SV01 server can be read by means of the redfish interface. Details regarding the functionality of the DM-SV01 sensors can be found in section “2.2.3 Sensors”.

3.3.5.1 Power Sensors

Using a GET request, it is possible to retrieve information about the power sensors from the DM-SV01 mainboard.

Function	Power sensors
Operation	GET
URI	https://<BMC_IP>/redfish/v1/Chassis/chassis/Power
Payload	none
Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Please, see the example below.

As an example, the excerpt below shows the data provided by the DM-SV01 mainboard using the redfish GET request. The user can check power and voltage measurements from all the power sensors available in the server.


```
{
   "@odata.id": "/redfish/v1/Chassis/chassis/Power",
   "@odata.type": "#Power.v1_5_2.Power",
   "Id": "Power",
   "Name": "Power",
   "PowerControl": [
      {
         "@odata.id": "/redfish/v1/Chassis/chassis/Power#/PowerControl/0",
         "@odata.type": "#Power.v1_0_0.PowerControl",
         "MemberId": "0",
         "Name": "Chassis Power Control",
         "PowerConsumedWatts": 5.691853999999999,
         "PowerLimit": {
            "LimitException": "NoAction",
            "LimitInWatts": null
         },
         "Status": {
            "Health": "OK",
            "State": "Enabled"
         }
      }
   ],
   "Redundancy": [],
   "Voltages": [
      {
         "@odata.id": "/redfish/v1/Chassis/chassis/Power#/Voltages/0",
         "@odata.type": "#Power.v1_0_0.Voltage",
         "LowerThresholdCritical": 10.8,
         "LowerThresholdNonCritical": 11.4,
         "MaxReadingRange": 0.0,
         "MemberId": "POWER_SUPPLY_IN",
         "MinReadingRange": 0.0,
         "Name": "POWER SUPPLY IN",
         "ReadingVolts": 12.255,
         "Status": {
            "Health": "OK",
            "State": "Enabled"
```

```
        },
        "UpperThresholdCritical": 13.200000000000001,
        "UpperThresholdNonCritical": 12.6
    },
    {
        "@odata.id": "/redfish/v1/Chassis/chassis/Power#/Voltages/1",
        "@odata.type": "#Power.v1_0_0.Voltage",
        "LowerThresholdCritical": 10.8,
        "LowerThresholdNonCritical": 11.4,
        "MaxReadingRange": 0.0,
        "MemberId": "POWER_SUPPLY_OUT",
        "MinReadingRange": 0.0,
        "Name": "POWER SUPPLY OUT",
        "ReadingVolts": 12.23,
        "Status": {
            "Health": "OK",
            "State": "Enabled"
        },
        "UpperThresholdCritical": 13.200000000000001,
        "UpperThresholdNonCritical": 12.6
    },
    {
        "@odata.id": "/redfish/v1/Chassis/chassis/Power#/Voltages/2",
        "@odata.type": "#Power.v1_0_0.Voltage",
        "LowerThresholdCritical": 2.97,
        "LowerThresholdNonCritical": 3.1350000000000002,
        "MaxReadingRange": 0.0,
        "MemberId": "VDD_33_DUAL",
        "MinReadingRange": 0.0,
        "Name": "VDD 33 DUAL",
        "ReadingVolts": 3.399,
        "Status": {
            "Health": "OK",
            "State": "Enabled"
        },
        "UpperThresholdCritical": 3.63,
        "UpperThresholdNonCritical": 3.465
    },
    {
        "@odata.id": "/redfish/v1/Chassis/chassis/Power#/Voltages/3",
        "@odata.type": "#Power.v1_0_0.Voltage",
        "LowerThresholdCritical": 2.97,
```

```
        "LowerThresholdNonCritical": 3.1350000000000002,
        "MaxReadingRange": 0.0,
        "MemberId": "VDD_33_RUN",
        "MinReadingRange": 0.0,
        "Name": "VDD 33 RUN",
        "ReadingVolts": 2.604,
        "Status": {
            "Health": "OK",
            "State": "Enabled"
        },
        "UpperThresholdCritical": 3.63,
        "UpperThresholdNonCritical": 3.465
    },
    {
        "@odata.id": "/redfish/v1/Chassis/chassis/Power#/Voltages/4",
        "@odata.type": "#Power.v1_0_0.Voltage",
        "LowerThresholdCritical": 4.5,
        "LowerThresholdNonCritical": 4.75,
        "MaxReadingRange": 0.0,
        "MemberId": "VDD_5_DUAL",
        "MinReadingRange": 0.0,
        "Name": "VDD 5 DUAL",
        "ReadingVolts": 5.204,
        "Status": {
            "Health": "OK",
            "State": "Enabled"
        },
        "UpperThresholdCritical": 5.5,
        "UpperThresholdNonCritical": 5.25
    }
  ]
}
```


Once the operation is successful, it returns the response “200 OK” and the sensors information is retrieved.

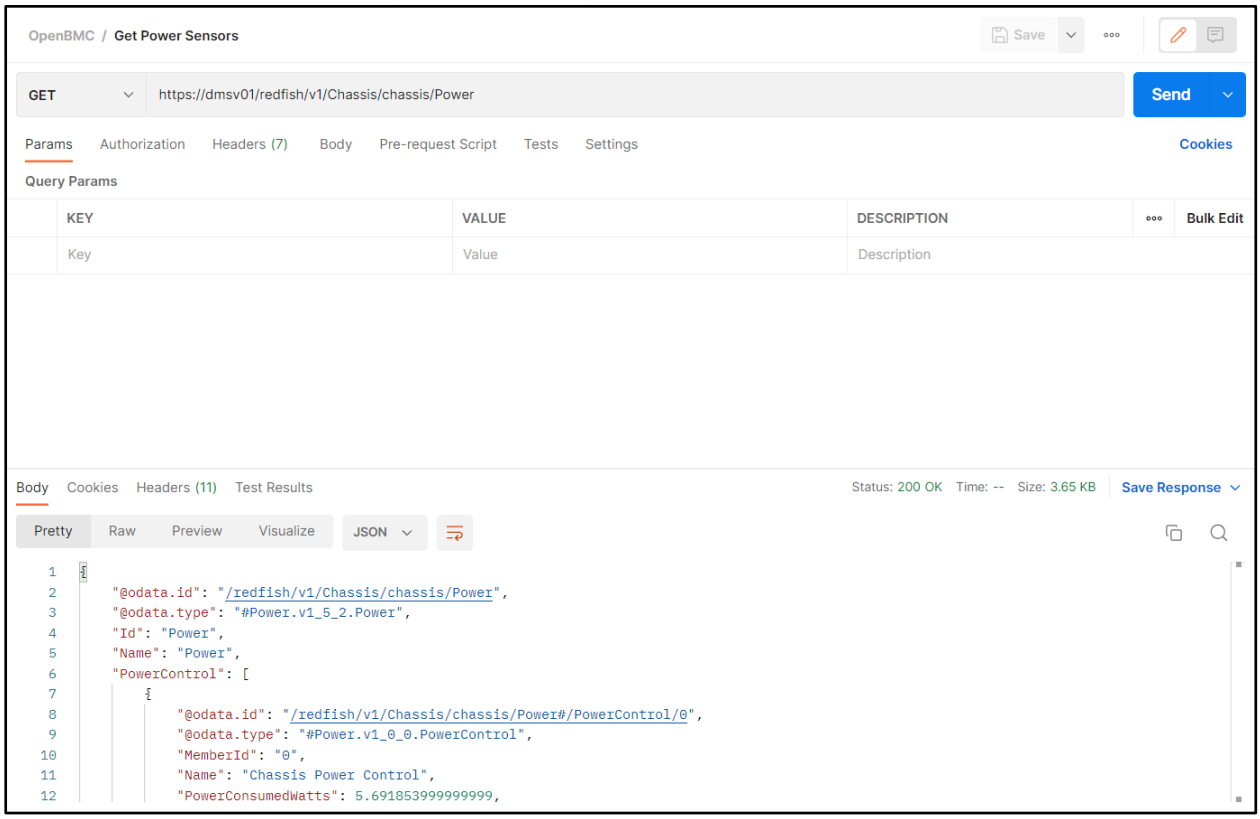


Figure 84: Redfish - Power Sensors

3.3.5.2 Temperature Sensors

Using a GET request, it is possible to retrieve information about the temperature sensors from the DM-SV01 mainboard.

Function	Temperature sensors
Operation	GET
URI	https://<BMC_IP>/redfish/v1/Chassis/chassis/Thermal
Payload	none
Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Please, see the example below.

As an example, the excerpt below shows the data provided by the DM-SV01 mainboard using the redfish GET request. The user can check the measurements from all the temperature sensors available in the server.

```
{
   "@odata.id": "/redfish/v1/Chassis/chassis/Thermal",
   "@odata.type": "#Thermal.v1_4_0.Thermal",
   "Fans": [
      {
         "@odata.id": "/redfish/v1/Chassis/chassis/Thermal#/Fans/0",
         "@odata.type": "#Thermal.v1_3_0.Fan",
         "LowerThresholdCritical": 1000,
         "LowerThresholdNonCritical": 1500,
         "MaxReadingRange": 0,
         "MemberId": "FAN_LEFT",
         "MinReadingRange": 0,
         "Name": "FAN LEFT",
         "Reading": 0,
         "ReadingUnits": "RPM",
         "Status": {
            "Health": "Critical",
            "State": "Enabled"
         },
         "UpperThresholdCritical": 13000,
         "UpperThresholdNonCritical": 12000
      },
[...]
   "Id": "Thermal",
   "Name": "Thermal",
   "Redundancy": [],
   "Temperatures": [
[...]
      {
         "@odata.id": "/redfish/v1/Chassis/chassis/Thermal#/Temperatures/2",
         "@odata.type": "#Thermal.v1_3_0.Temperature",
         "LowerThresholdCritical": 0.0,
         "LowerThresholdNonCritical": 5.0,
         "MaxReadingRangeTemp": 0.0,
         "MemberId": "INLET",
         "MinReadingRangeTemp": 0.0,
         "Name": "INLET",
         "ReadingCelsius": 25.313,
         "Status": {
            "Health": "OK",
            "State": "Enabled"
         },
```

```
        "UpperThresholdCritical": 50.0,
        "UpperThresholdNonCritical": 40.0
    },
[...]
```


Once the operation is successful, it returns the response “200 OK” and the sensors information is retrieved.

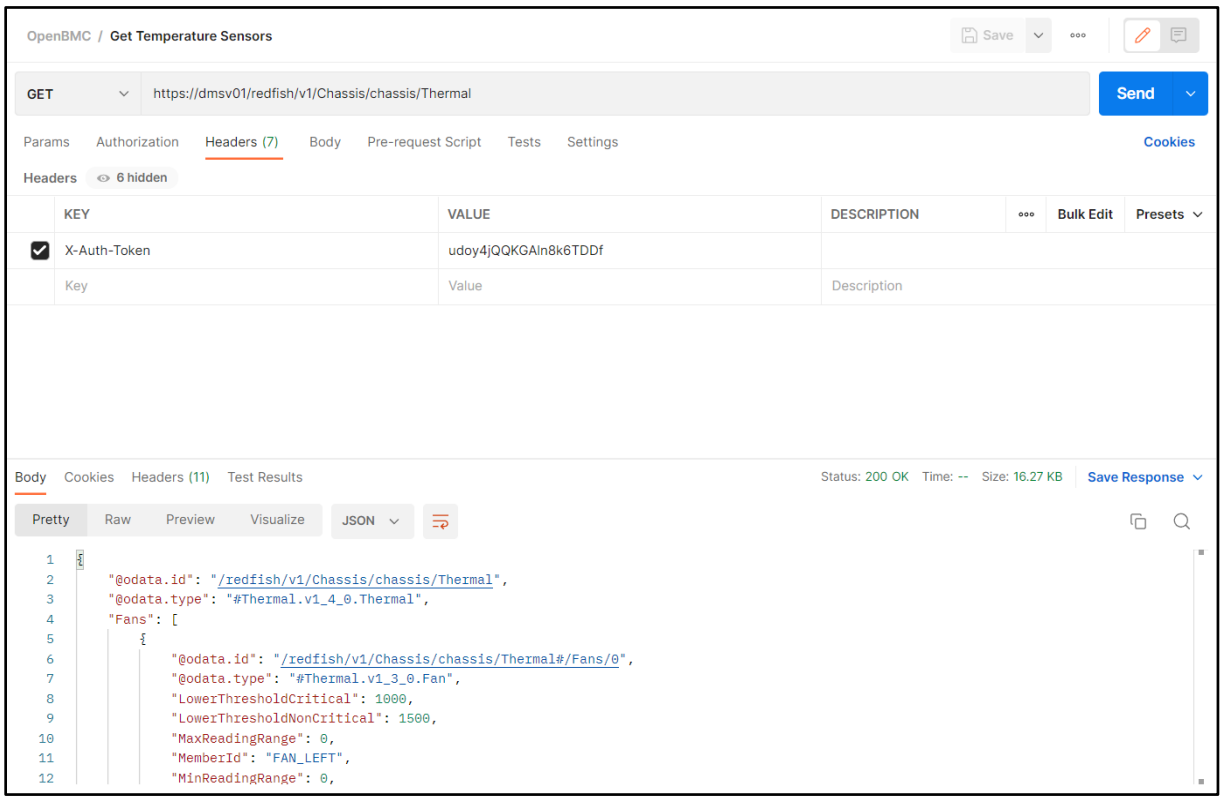


Figure 85: Redfish - Temperature Sensors

3.3.5.3 Power Consumption sensors

3.3.5.3.1 Current Consumption

Using a GET request, it is possible to retrieve the instantaneous total current being consumed by the DM-SV01 server.

Function	System Total Current Consumption
Operation	GET
URI	https://<BMC_IP>/redfish/v1/Chassis/chassis/Sensors/POWER_SUPPLY
Payload	none
Header	X-Auth-Token: “<token>”

Expected response	200 OK
Reply	Please, see the example below.

As an example, the excerpt below shows the data provided by the redfish when using the GET request to retrieve the total current consumption of the system. The value of the current consumption is shown in the “Reading” field. The measurement unit (Amperes) is shown in the “ReadingUnits” field. Additionally, the user can check the health status of the sensor and the respective warning and critical thresholds.


```
{
   "@odata.id": "/redfish/v1/Chassis/chassis/Sensors/POWER_SUPPLY",
   "@odata.type": "#Sensor.v1_0_0.Sensor",
   "Id": "POWER_SUPPLY",
   "Name": "POWER SUPPLY",
   "Reading": 5.394,
   "ReadingRangeMax": 0.0,
   "ReadingRangeMin": 0.0,
   "ReadingUnits": "Amperes",
   "Status": {
      "Health": "OK",
      "State": "Enabled"
   },
   "Thresholds": {
      "LowerCaution": {
         "Reading": 0.0
      },
      "LowerCritical": {
         "Reading": 0.0
      },
      "UpperCaution": {
         "Reading": 61.0
      },
      "UpperCritical": {
         "Reading": 65.0
      }
   }
}
```


Once the operation is successful, it returns the response “200 OK” and the sensor information is retrieved.

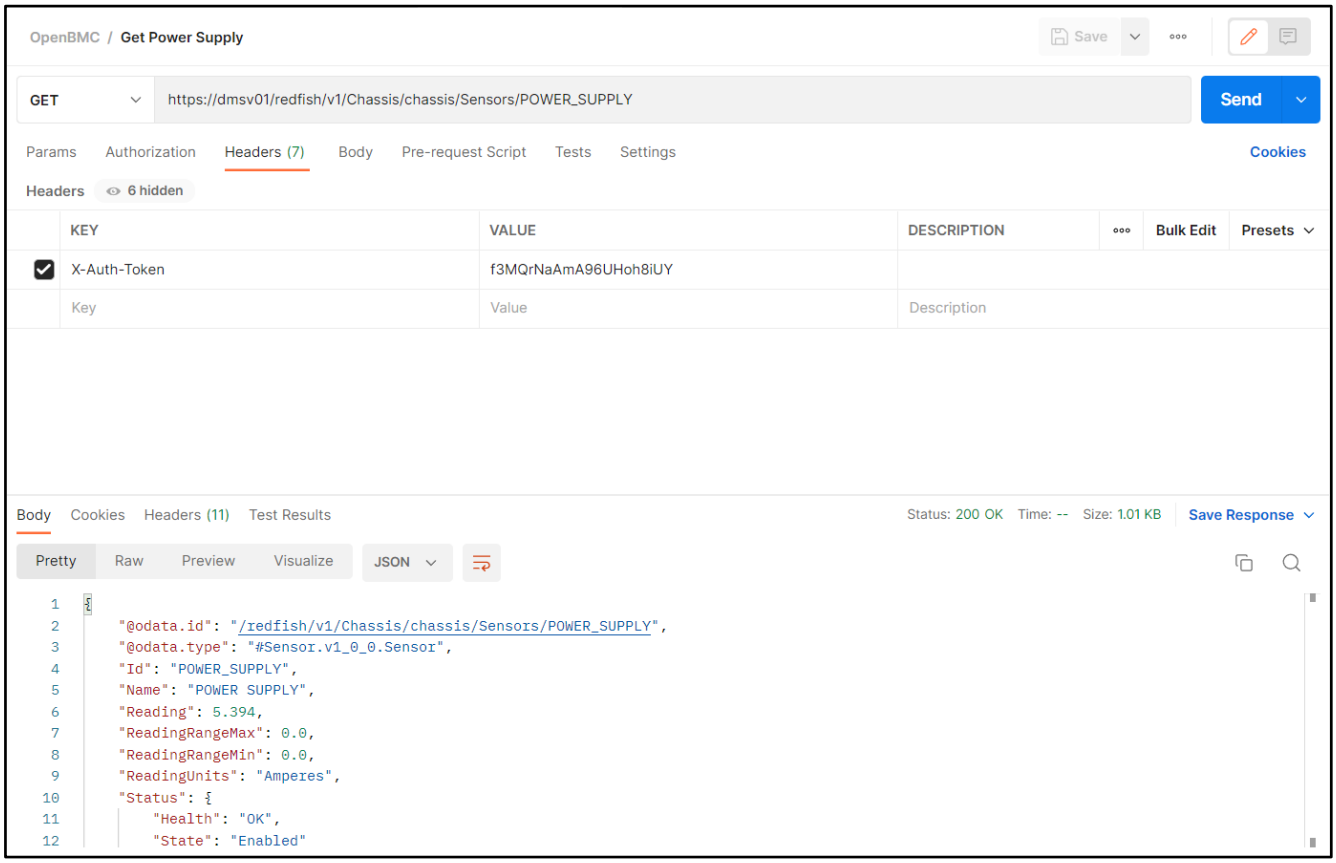


Figure 86: Redfish - Current consumption

3.3.5.3.2 Power Consumption

Using a GET request, it is possible to retrieve the instantaneous total power being consumed by the DM-SV01 server.

Function	System Total Power Consumption
Operation	GET
URI	https://<BMC_IP>/redfish/v1/Chassis/chassis/Sensors/total_power
Payload	none
Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Please, see the example below.

As an example, the excerpt below shows the data provided by the redfish when using the GET request to retrieve the total power consumption of the system. The value of the power consumption is shown in the “Reading” field. The measurement unit (Watts) is shown in the “ReadingUnits” field. Additionally, the user can check the health status of the sensor and the respective warning and critical thresholds.

```
{
   "@odata.id": "/redfish/v1/Chassis/chassis/Sensors/total_power",
   "@odata.type": "#Sensor.v1_0_0.Sensor",
   "Id": "total_power",
   "Name": "total power",
   "Reading": 4.5632969999999995,
   "ReadingRangeMax": 0.0,
   "ReadingRangeMin": 0.0,
   "ReadingUnits": "Watts",
   "Status": {
      "Health": "OK",
      "State": "Enabled"
   },
   "Thresholds": {
      "LowerCaution": {
         "Reading": 0.0
      },
      "LowerCritical": {
         "Reading": 0.0
      },
      "UpperCaution": {
         "Reading": 732.0
      },
      "UpperCritical": {
         "Reading": 780.0
      }
   }
}
```


Once the operation is successful, it returns the response “200 OK” and the sensor information is retrieved.

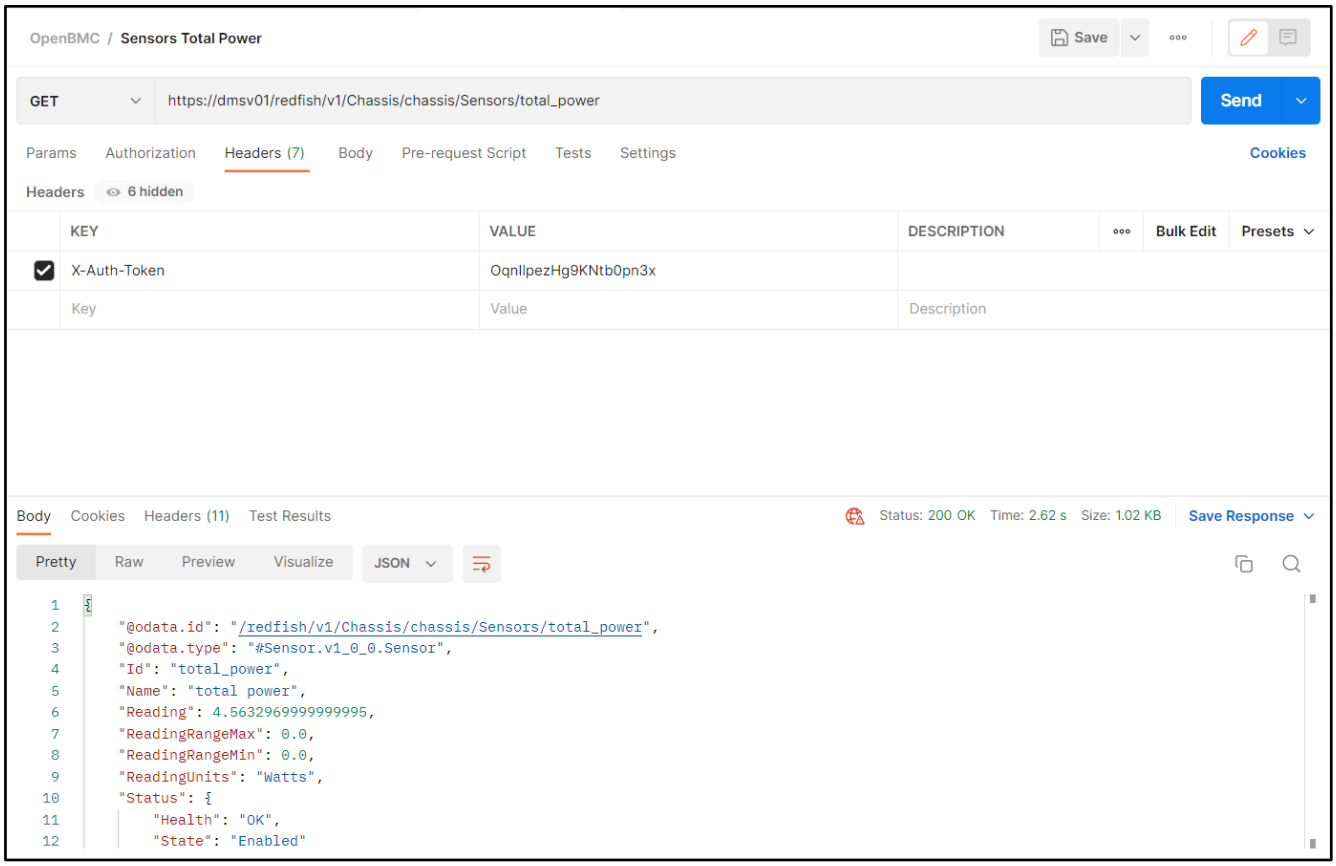


Figure 87: Redfish - Power consumption

3.3.5.3.3 Peak Power

Using a GET request, it is possible to retrieve the peak power consumed by the DM-SV01 server.

Function	System Peak Power
Operation	GET
URI	https://<BMC_IP>/redfish/v1/Chassis/chassis/Sensors/PEAK_POWER
Payload	none
Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Please, see the example below.

As an example, the excerpt below shows the data provided by the redfish when using the GET request to retrieve the peak power of the system. The value of the peak power is shown in the “Reading” field. The measurement unit (Watts) is shown in the “ReadingUnits” field. Additionally, the user can check the health status of the sensor and the respective warning and critical thresholds.

```
{
   "@odata.id": "/redfish/v1/Chassis/chassis/Sensors/PEAK_POWER",
   "@odata.type": "#Sensor.v1_0_0.Sensor",
   "Id": "PEAK_POWER",
   "Name": "PEAK POWER",
   "Reading": 99.263984,
   "ReadingRangeMax": 0.0,
   "ReadingRangeMin": 0.0,
   "ReadingUnits": "Watts",
   "Status": {
      "Health": "OK",
      "State": "Enabled"
   },
   "Thresholds": {
      "LowerCaution": {
         "Reading": 0.0
      },
      "LowerCritical": {
         "Reading": 0.0
      },
      "UpperCaution": {
         "Reading": 732.0
      },
      "UpperCritical": {
         "Reading": 780.0
      }
   }
}
```


Once the operation is successful, it returns the response “200 OK” and the sensor information is retrieved.

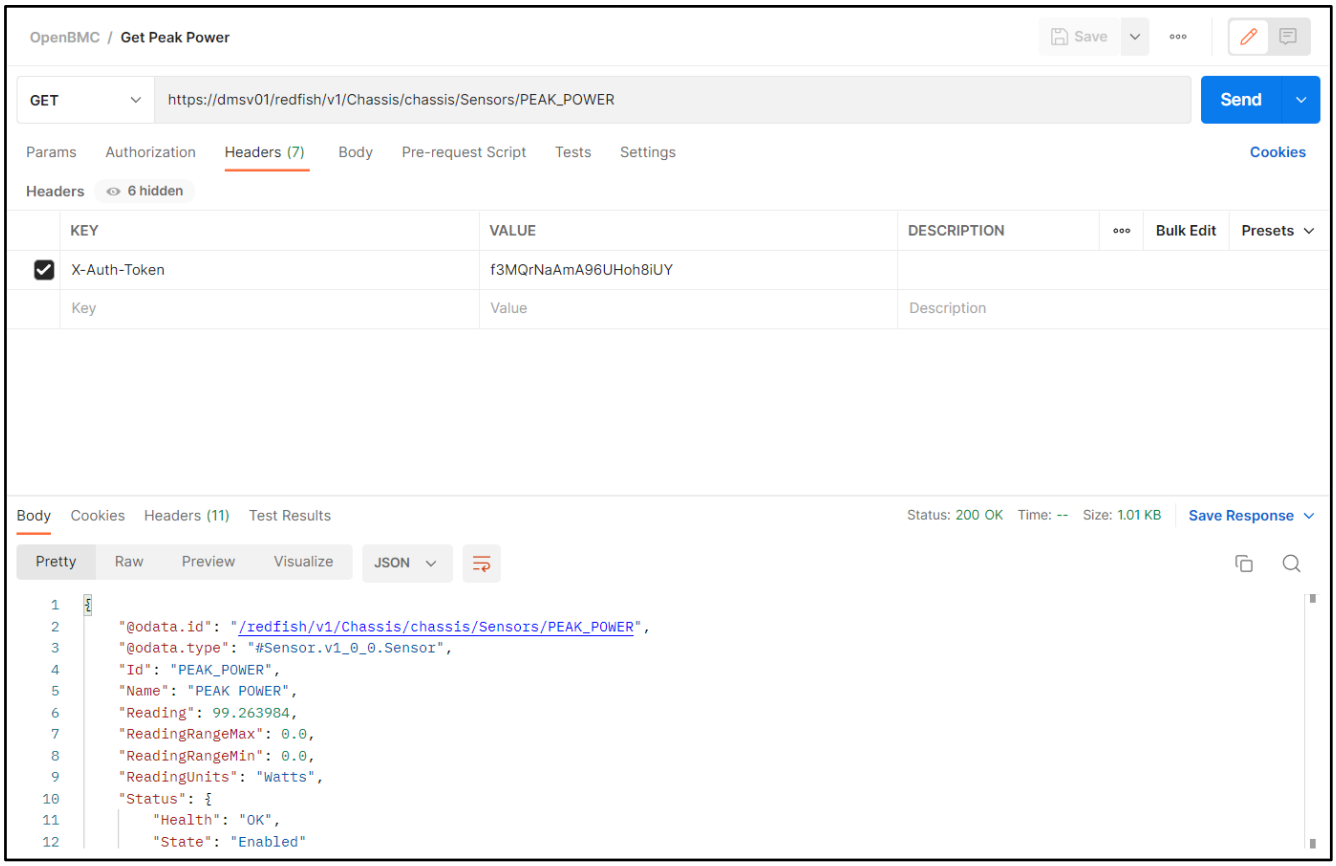


Figure 88: Redfish - Peak Power

Important: if the “Peak Power” sensor is not available in the BMC, please update the BMC SW by following the procedure described in section 2.4.2.2 FW update process - BMC or BIOS.

3.3.5.4 Peak Power sensor reset

Using a POST request, it is possible to reset the Peak Power Sensor.

Function	Reset Peak Power sensor
Operation	POST
URI	https://<BMC_IP>/redfish/v1/Managers/bmc/Oem/Datacom/Actions/Manager.ResetPeakPowerSensor

Payload	{
	"ResetToDefaultsType": "ResetAll"
	}

Header	X-Auth-Token: “<token>”
Expected response	200 OK

Reply	{
	"@Message.ExtendedInfo": [
	{
	"@odata.type": "#Message.v1_0_0.Message",
	"Message": "Successfully Completed Request",
	"MessageArgs": [],
	"MessageId": "Base.1.4.0.Success",
	"Resolution": "None",
	"Severity": "OK"
	}
	]
	}

Once the operation is successful, it returns the response “200 OK” and the peak power sensor reset is performed.

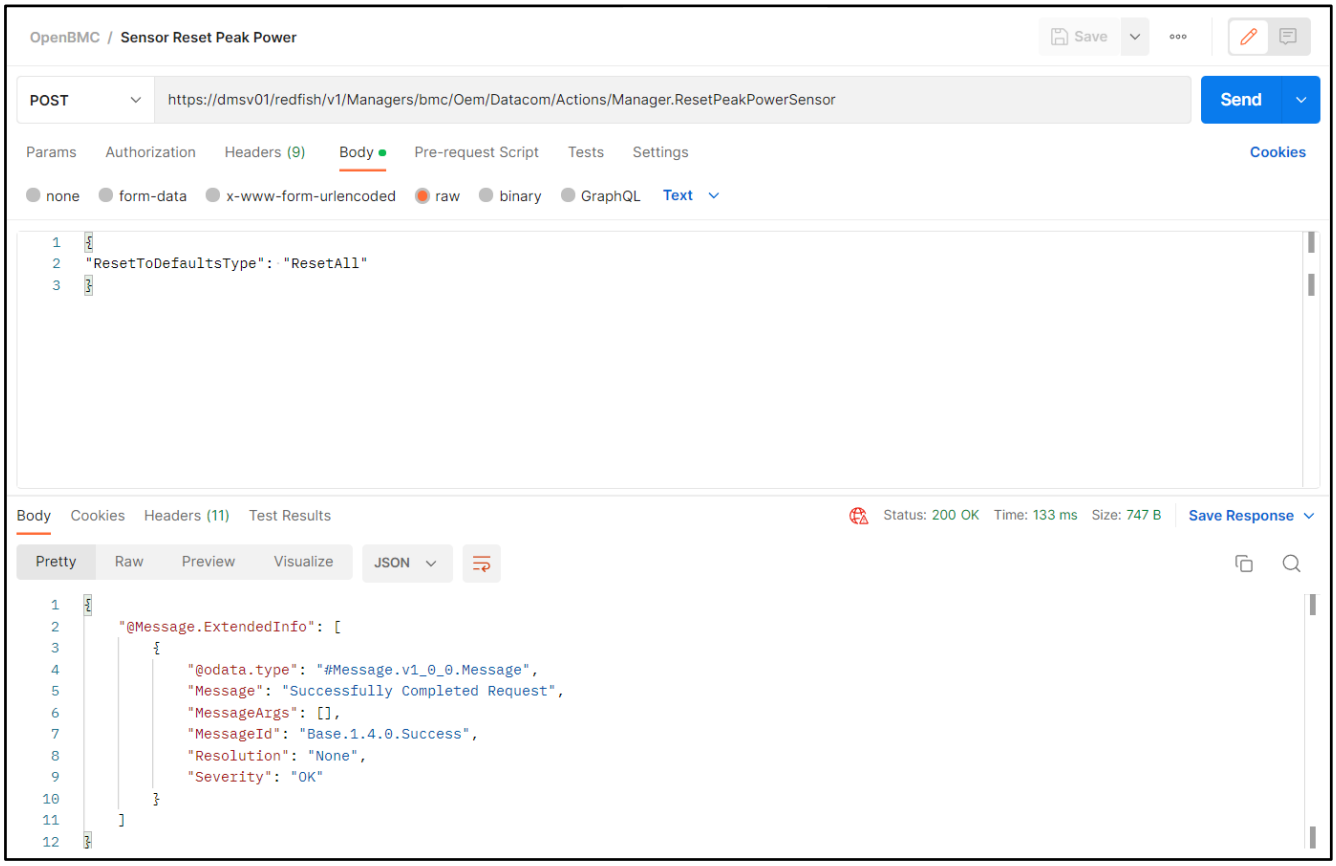


Figure 89: Redfish - Reset Peak Power Sensor

Important: if the “Peak Power” sensor is not available in the BMC, please update the BMC SW by following the procedure described in section 2.4.2.2 FW update process - BMC or BIOS.

3.3.5.5 Energy sensor reset

Using a POST request, it is possible to reset the Energy Sensor.

Function	Reset Energy sensor
Operation	POST
URI	https://<BMC_IP>/redfish/v1/Managers/bmc/Oem/Datacom/Actions/Manager.ResetTotalEnergySensor

Payload	{
	"ResetToDefaultsType": "ResetAll"
	}

Header	X-Auth-Token: “<token>”
Expected response	200 OK

Reply	{
	"@Message.ExtendedInfo": [
	{
	"@odata.type": "#Message.v1_0_0.Message",
	"Message": "Successfully Completed Request",
	"MessageArgs": [],
	"MessageId": "Base.1.4.0.Success",
	"Resolution": "None",
	"Severity": "OK"
	}
	]
	}

Once the operation is successful, it returns the response “200 OK” and the energy sensor reset is performed.

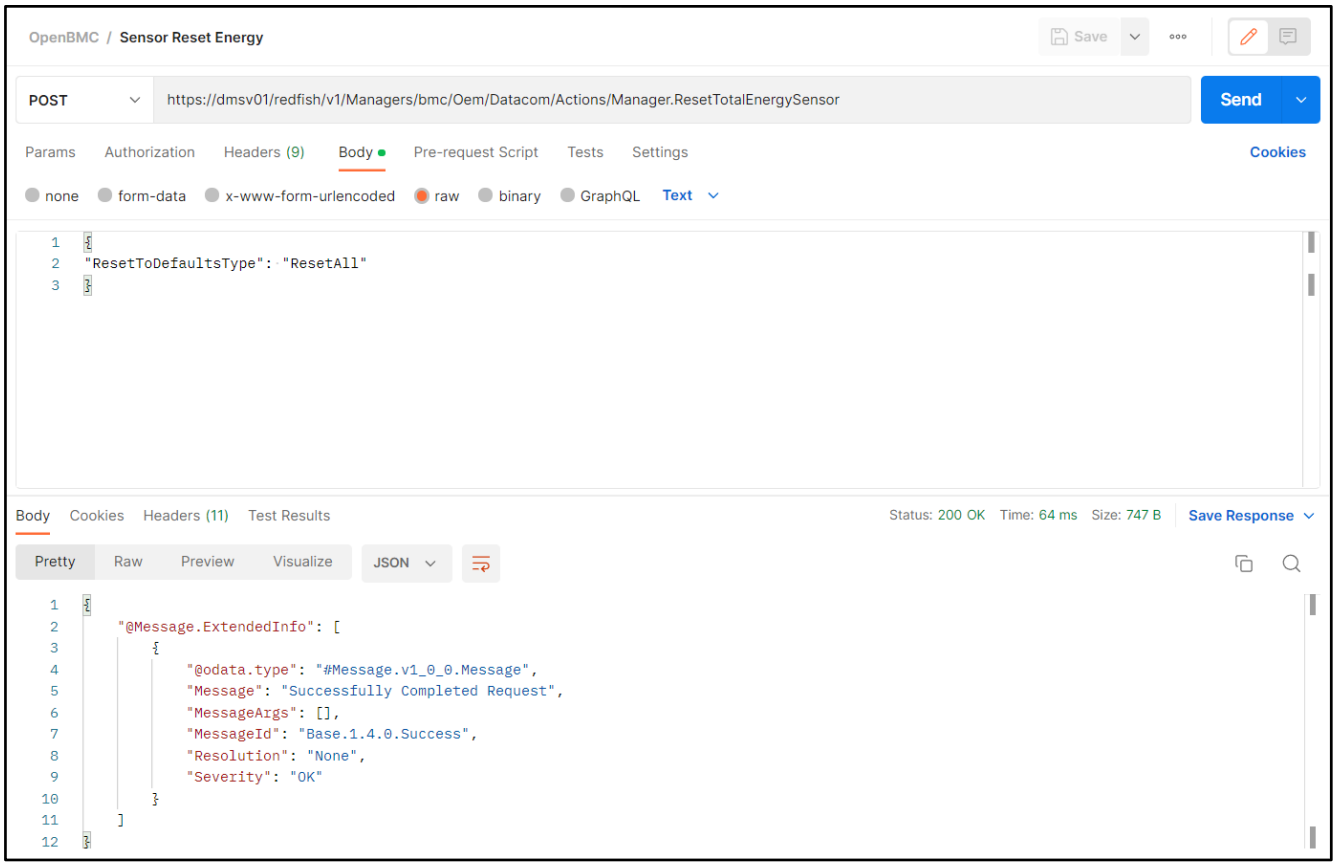


Figure 90: Redfish - Reset Energy Sensor

Important: if the “Energy” sensor is not available in the BMC, please update the BMC SW by following the procedure described in section 2.4.2.2 FW update process - BMC or BIOS.

3.3.6 Indicator LED

The indicator LED can be turned on or off by means of the redfish interface. Details regarding the functionality and use cases of the Indicator LED can be found in section “2.3.2 Server LED”.

3.3.6.1 Turn on Indicator LED

Using a PATCH request, it is possible to turn the LED indicator on.

Function	Turn on Server Indicator LED
Operation	PATCH
URI	https://<BMC_IP>/redfish/v1/Systems/system

Payload	{
	"IndicatorLED": "Lit"
	}

Header	X-Auth-Token: “<token>”
Expected response	204 No Content

Reply	None

Once the operation is successful, it returns the response “204 No Content” and the LED is activated.

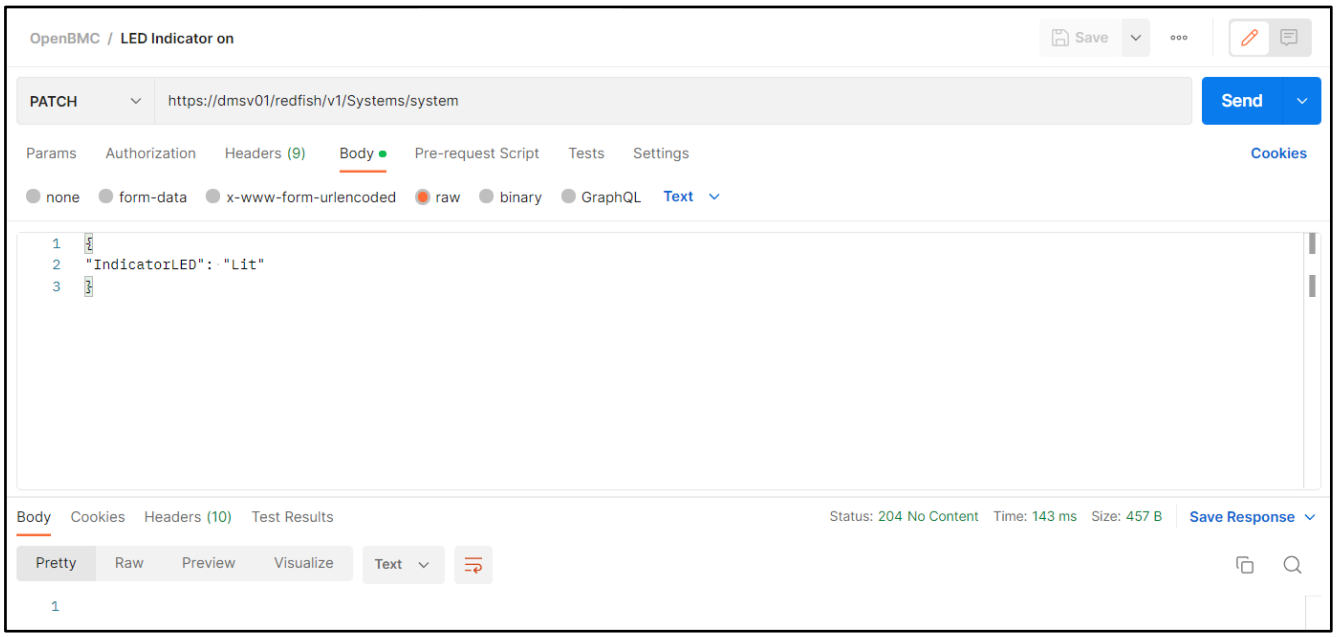


Figure 91: Redfish - LED Indicator turned on

3.3.6.2 Turn off Indicator LED

Using a PATCH request, it is possible to turn the LED indicator off.

Function	Turn off Server Indicator LED
Operation	PATCH
URI	https://<BMC_IP>/redfish/v1/Systems/system

Payload	{
	"IndicatorLED": "Off"
	}

Header	X-Auth-Token: “<token>”
Expected response	204 No Content
Reply	None

Once the operation is successful, it returns the response “204 No Content” and the LED is deactivated.

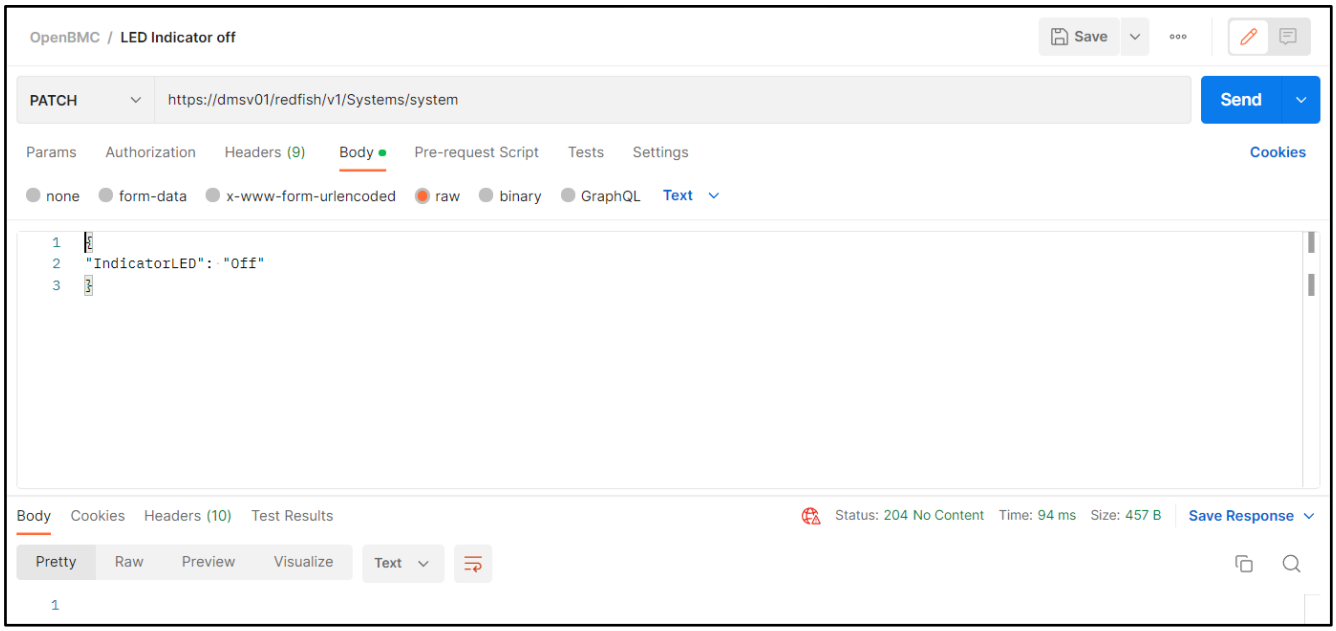


Figure 92: Redfish - LED Indicator turned off

3.3.7 Host Power Actions

The host processors can have their power controlled by means of the redfish. Details regarding the power functions for the host processors can be found in section “2.3.1.1 Operations”.

3.3.7.1 Power On Host

Using a POST request, it is possible to power the host processors on.

Function	Host power on
Operation	POST
URI	https://<BMC_IP>/redfish/v1/Systems/system/Actions/ComputerSystem.Reset

Payload	{
	"ResetType": "On"
	}

Header	X-Auth-Token: “<token>”
Expected response	200 OK

Reply	{
	"@Message.ExtendedInfo": [
	{
	"@odata.type": "#Message.v1_0_0.Message",
	"Message": "Successfully Completed Request",
	"MessageArgs": [],
	"MessageId": "Base.1.4.0.Success",
	"Resolution": "None",
	"Severity": "OK"
	}
	]
	}

Once the operation is successful, it returns the response “200 OK” and the host processors are powered on.

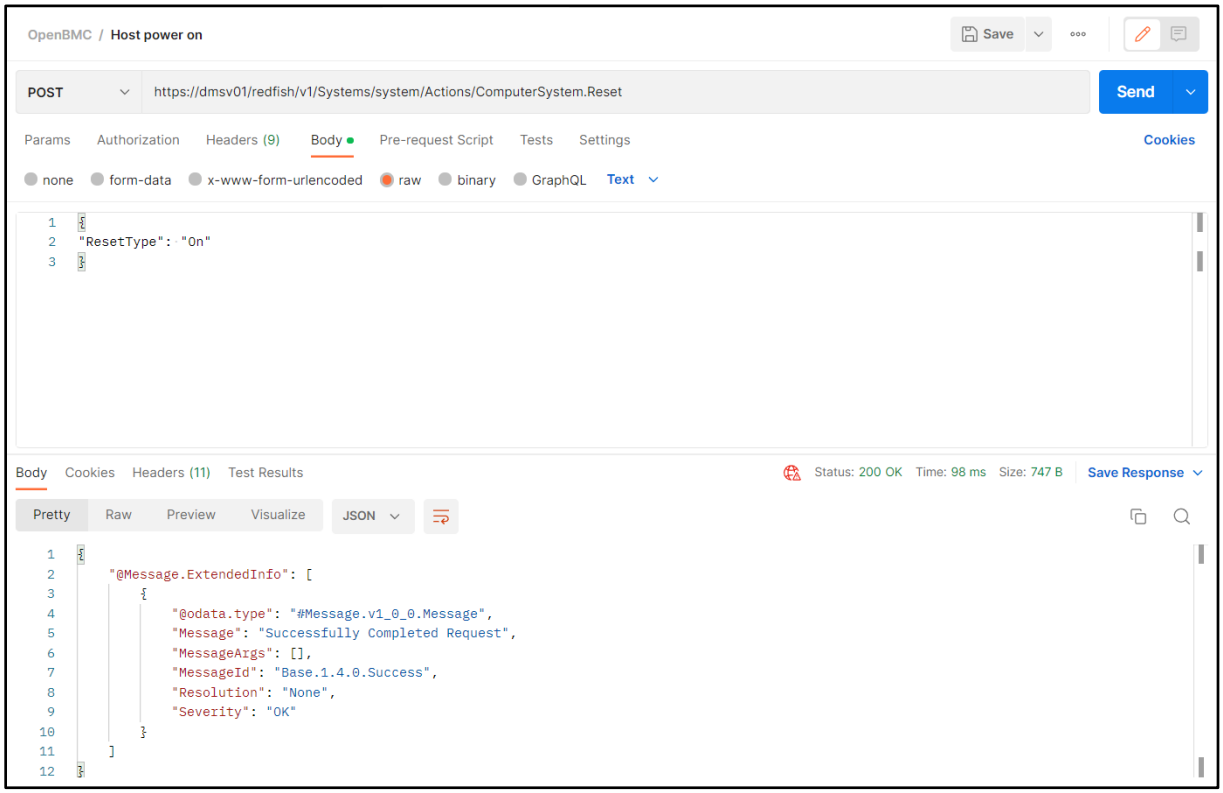


Figure 93: Redfish - Host power on

3.3.7.2 Power Off Host

Using a POST request, it is possible to turn the host processors off.

Function	Host power off
Operation	POST

URI	https://<BMC_IP>/redfish/v1/Systems/system/Actions/ComputerSystem.Reset

Payload	{
	"ResetType": "GracefulShutdown"
	}

Header	X-Auth-Token: “<token>”
Expected response	200 OK

Reply	{
	"@Message.ExtendedInfo": [
	{
	"@odata.type": "#Message.v1_0_0.Message",
	"Message": "Successfully Completed Request",
	"MessageArgs": [],
	"MessageId": "Base.1.4.0.Success",
	"Resolution": "None",
	"Severity": "OK"
	}
	]
	}

Once the operation is successful, it returns the response “200 OK” and the host processors are powered off.

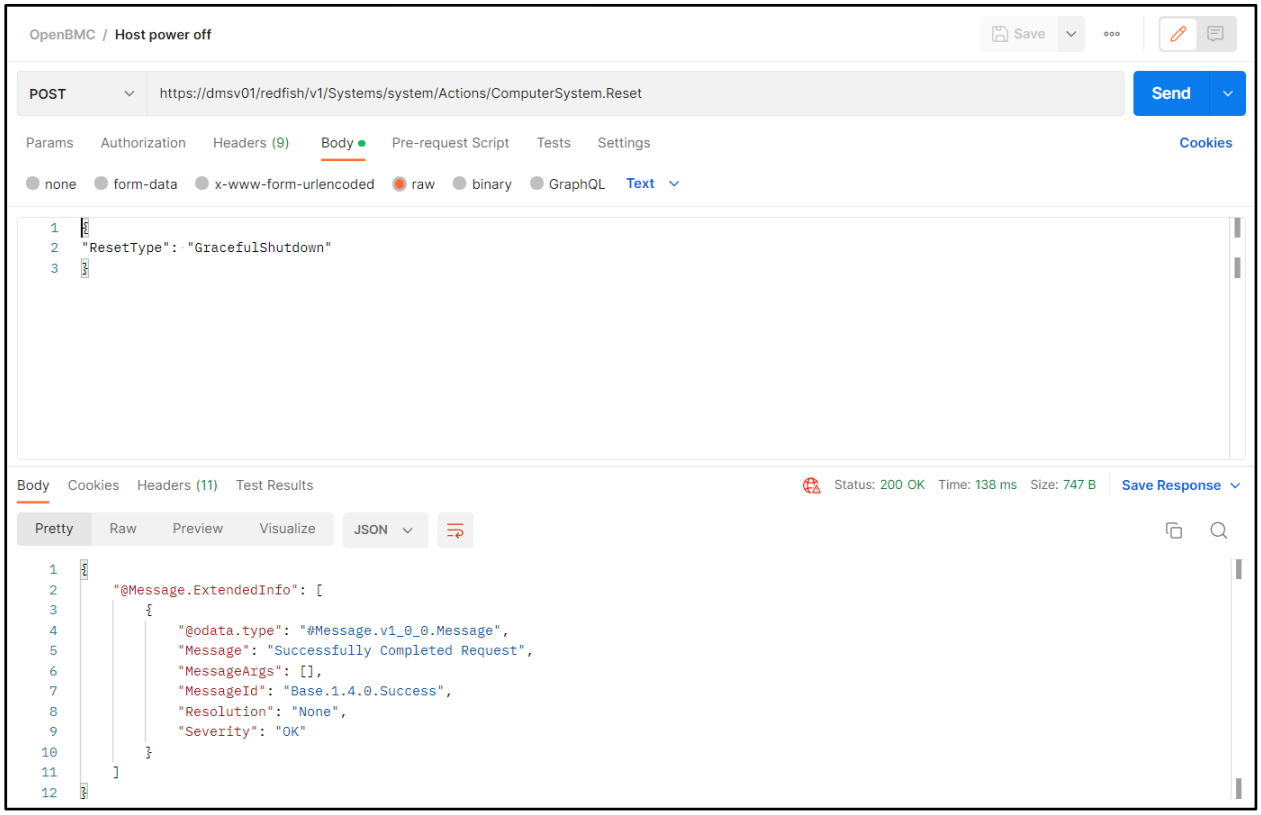


Figure 94: Redfish - Host power off

3.3.7.3 Restart Host

Using a POST request, it is possible to restart the host processors.

Function	Host restart
Operation	POST
URI	https://<BMC_IP>/redfish/v1/Systems/system/Actions/ComputerSystem.Reset

Payload	{
	"ResetType": "GracefulRestart"
	}

Header	X-Auth-Token: “<token>”
Expected response	200 OK

Reply	{
	"@Message.ExtendedInfo": [
	{
	"@odata.type": "#Message.v1_0_0.Message",
	"Message": "Successfully Completed Request",
	"MessageArgs": [],
	"MessageId": "Base.1.4.0.Success",
	"Resolution": "None",
	"Severity": "OK"
	}
	]
	}

Once the operation is successful, it returns the response “200 OK” and the host processors are restarted.

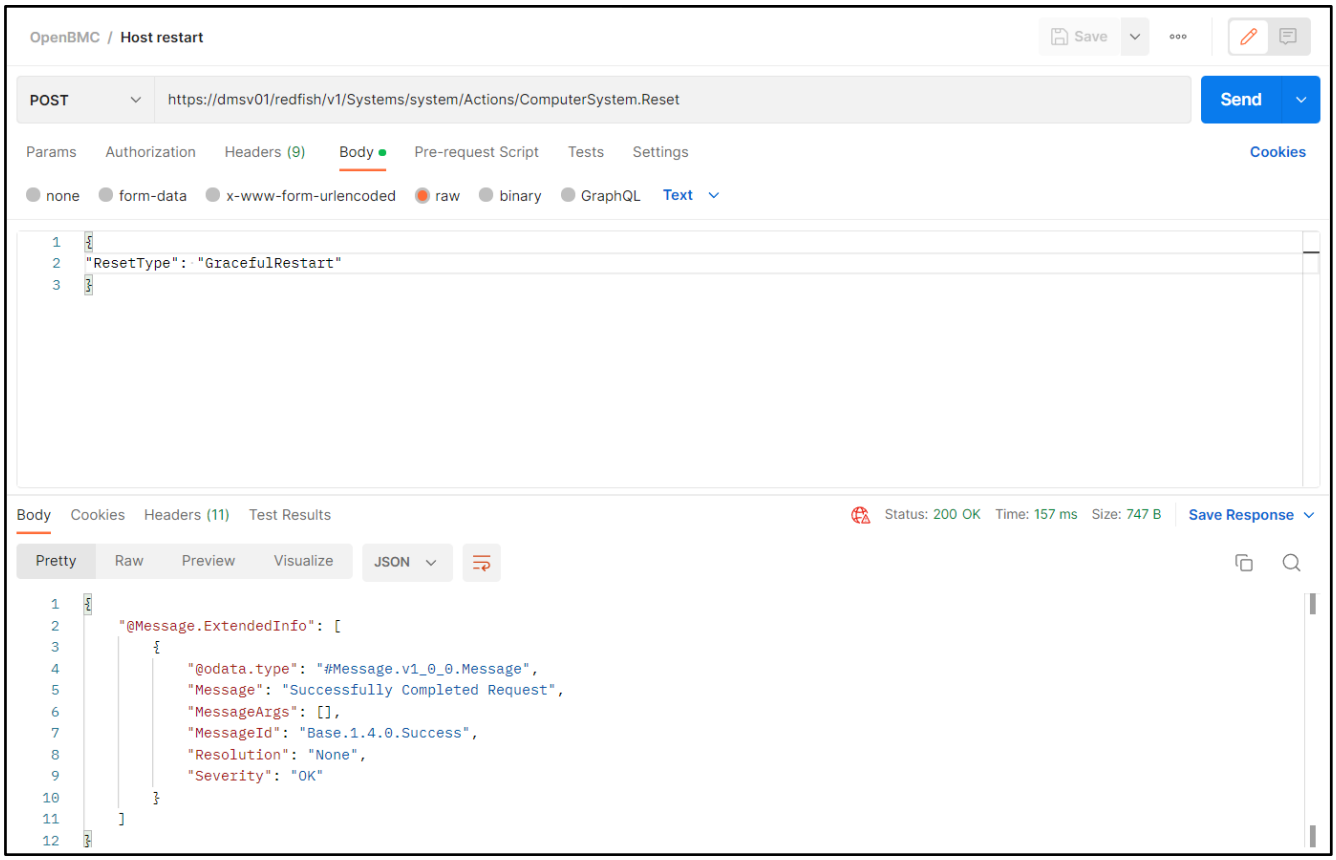


Figure 95: Redfish - Host restart

3.3.7.4 Force Power Off Host

Using a POST request, it is possible to forcibly turn the host processors off.

Function	Host forced power off
Operation	POST
URI	https://<BMC_IP>/redfish/v1/Systems/system/Actions/ComputerSystem.Reset

Payload	{
	"ResetType": "ForceOff"
	}

Header	X-Auth-Token: “<token>”
Expected response	200 OK

Reply	{
	"@Message.ExtendedInfo": [
	{
	"@odata.type": "#Message.v1_0_0.Message",
	"Message": "Successfully Completed Request",
	"MessageArgs": [],
	"MessageId": "Base.1.4.0.Success",
	"Resolution": "None",
	"Severity": "OK"
	}
	]
	}

Once the operation is successful, it returns the response “200 OK” and the host processors are forcibly powered off.

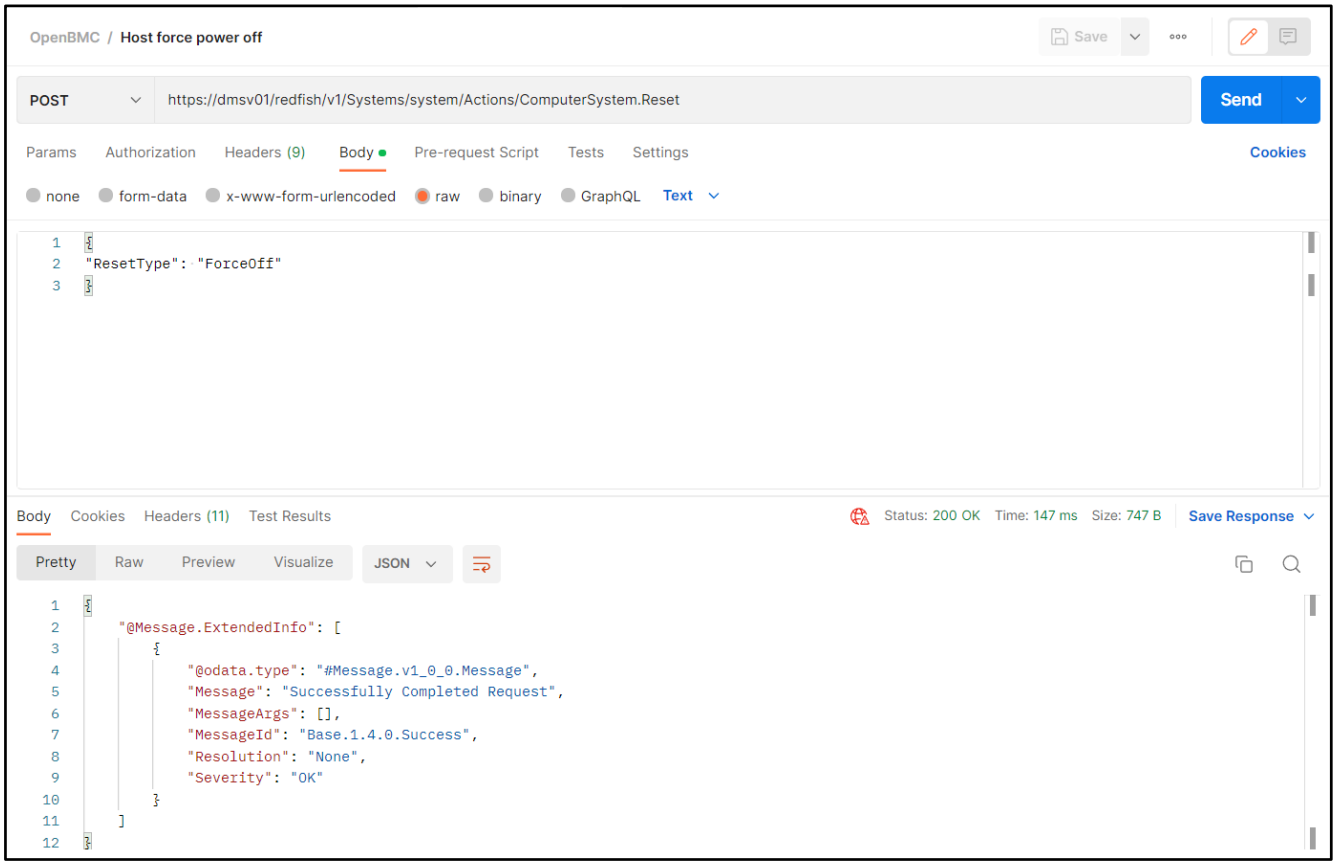


Figure 96: Redfish - Host force power off

3.3.7.5 Force Restart Host

Using a POST request, it is possible to forcibly restart the host processors.

Function	Host forced restart
Operation	POST
URI	https://<BMC_IP>/redfish/v1/Systems/system/Actions/ComputerSystem.Reset

Payload	{
	"ResetType": "ForceRestart"
	}

Header	X-Auth-Token: “<token>”
Expected response	200 OK

Reply	{
	"@Message.ExtendedInfo": [
	{
	"@odata.type": "#Message.v1_0_0.Message",
	"Message": "Successfully Completed Request",
	"MessageArgs": [],
	"MessageId": "Base.1.4.0.Success",
	"Resolution": "None",
	"Severity": "OK"
	}
	]
	}

Once the operation is successful, it returns the response “200 OK” and the host processors are forcibly restarted.

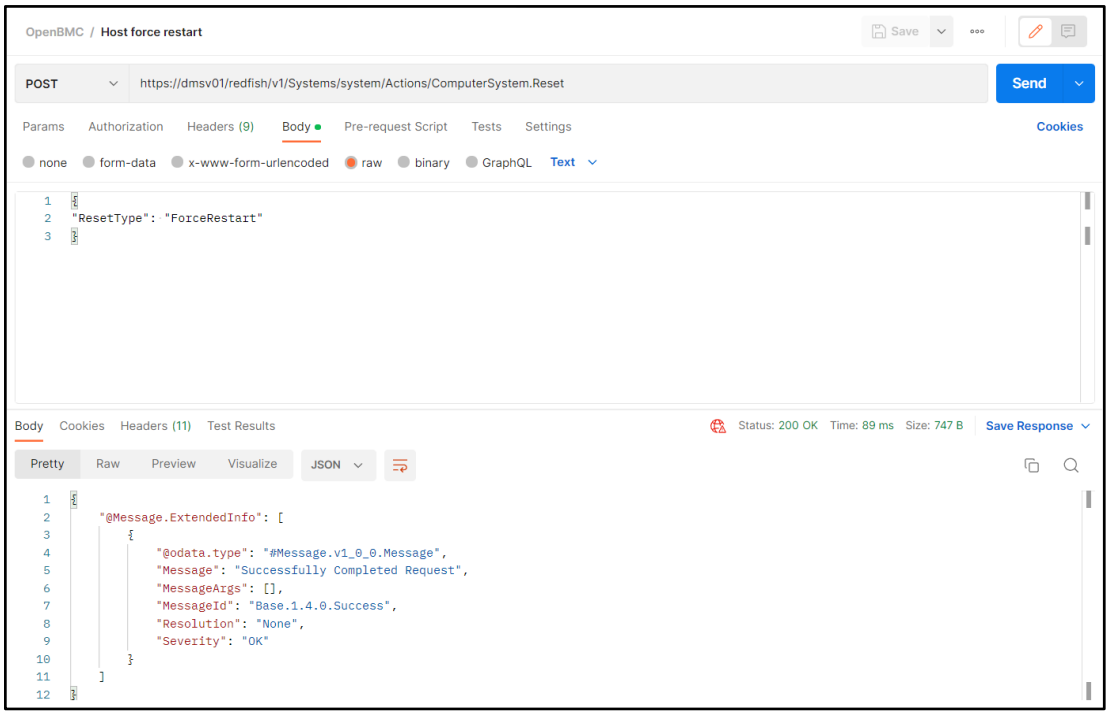


Figure 97: Redfish - Host force restart

3.3.8 Network Settings

Using a PATCH request, it is possible to configure the network settings of the BMC. The user can configure both inband and out of band ports by using the correct alias in the URI of the request, as follows:

- **“eth0”:** this interface is the “NC-SI” (Network Controller Sideband Interface). The NC-SI interface is used for inband management of the BMC. The eth0 interface is accessed by means of the mezzanine card Ethernet port 0.
- **“eth1”:** it is the default out-of-band management interface of the BMC. It can be accessed by means of the dedicated Ethernet port present in the front panel of the DM-SV01.

The configuration parameters are inserted in the payload of the request, as shown in the example from the table below. Details regarding the network settings of the BMC can be found in section “2.4.1 Network settings”.

Function	Network Settings
Operation	PATCH
URI	https://<BMC_IP>/redfish/v1/Managers/bmc/EthernetInterfaces/eth1

Payload	{
	"HostName": "dmsv01",
	"IPv4StaticAddresses": [
	{
	"Address": "192.168.15.101",
	"Gateway": "192.168.15.1",
	"SubnetMask": "255.255.255.0"
	}
	],
	"StaticNameServers": ["8.8.8.8"]
	}

Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	None

Once the operation is successful, it returns the response “200 OK” and the network settings are applied according to the payload.

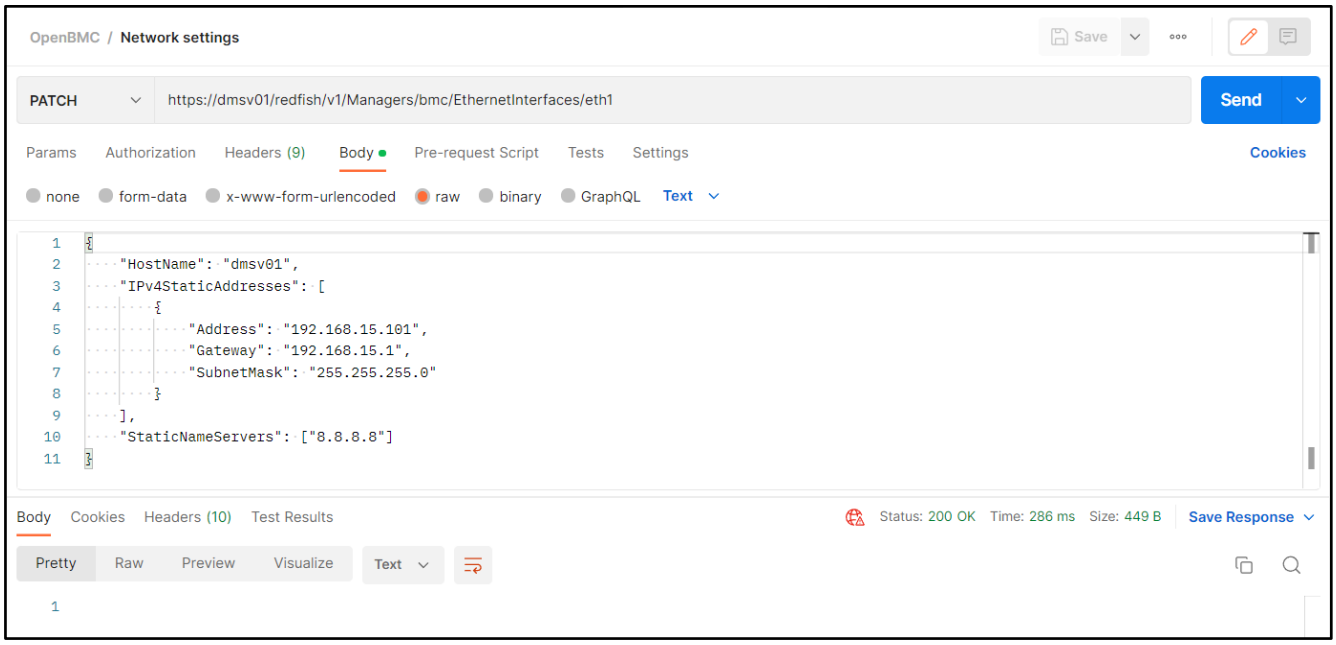


Figure 98: Redfish - Configuring network

3.3.9 Boot Override Options

The user can force the boot sequence configuration by means of the redfish. Details regarding the boot override function can be found in section “2.3.1.2 Host OS boot settings - Boot override”.

3.3.9.1 Force PXE Boot Override

Using a PATCH request, it is possible to configure the override function to force the system to boot from PXE (netboot).

Function	Boot override - PXE
Operation	PATCH
URI	https://<BMC_IP>/redfish/v1/Systems/system

Payload	{
	“Boot”: {
	“BootSourceOverrideTarget” : “Pxe”
	}
	}

Header	X-Auth-Token: “<token>”
Expected response	204 No Content
Reply	None

Once the operation is successful, it returns the response “204 No Content” and the host system will boot by means of the PXE.

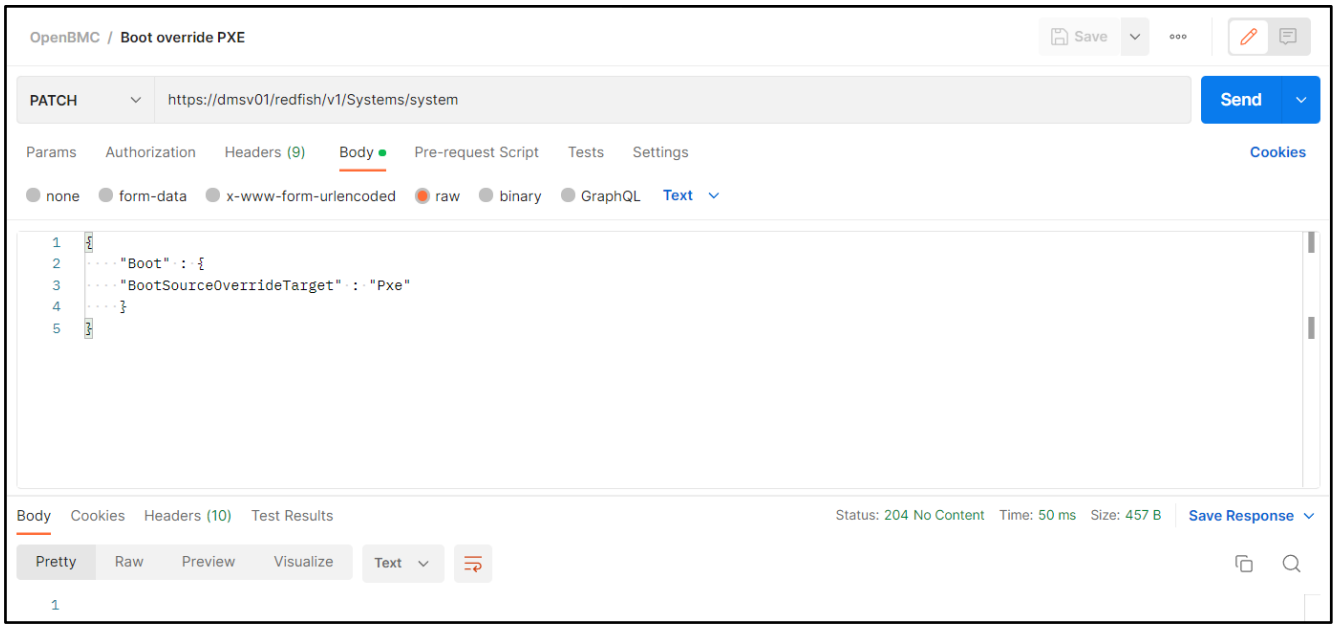


Figure 99: Redfish - Boot override option configured as PXE

3.3.9.2 Force CD-ROM/Virtual Media Boot Override

Using a PATCH request, it is possible to configure the override function to force the system to boot from CD (also used for booting the virtual media).

Function	Boot override - CD
Operation	PATCH
URI	https://<BMC_IP>/redfish/v1/Systems/system

Payload	{
	“Boot”: {
	“BootSourceOverrideTarget” : “Cd”
	}
	}

Header	X-Auth-Token: “<token>”
Expected response	204 No Content
Reply	None

Once the operation is successful, it returns the response “204 No Content” and the host system will boot by means of the CD or virtual media.

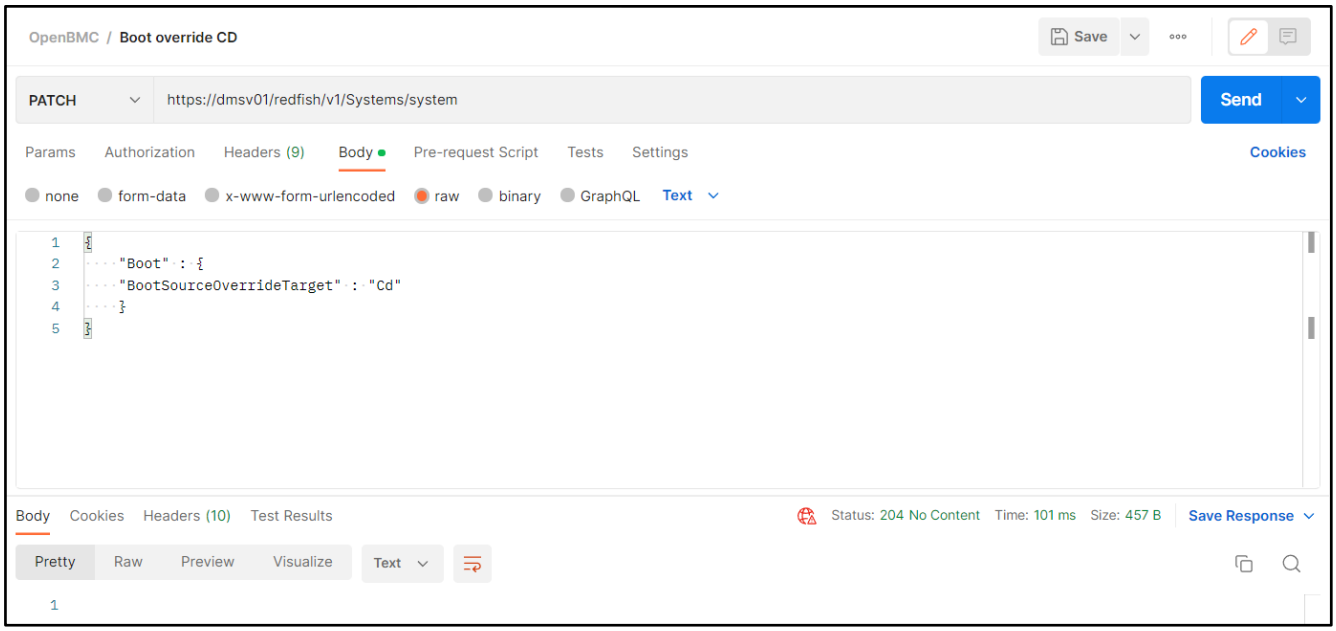


Figure 100: Redfish - Boot override option configured as CD

3.3.9.3 Force BIOS Setup Boot Override

Using a PATCH request, it is possible to configure the override function to force the system to boot the BIOS/UEFI setup.

Function	Boot override - BIOS/UEFI setup
Operation	PATCH
URI	https://<BMC_IP>/redfish/v1/Systems/system

Payload	{
	“Boot”: {
	“BootSourceOverrideTarget” : “BiosSetup”
	}
	}

Header	X-Auth-Token: “<token>”
Expected response	204 No Content
Reply	None

Once the operation is successful, it returns the response “204 No Content” and the host system will boot the BIOS/UEFI setup screen.

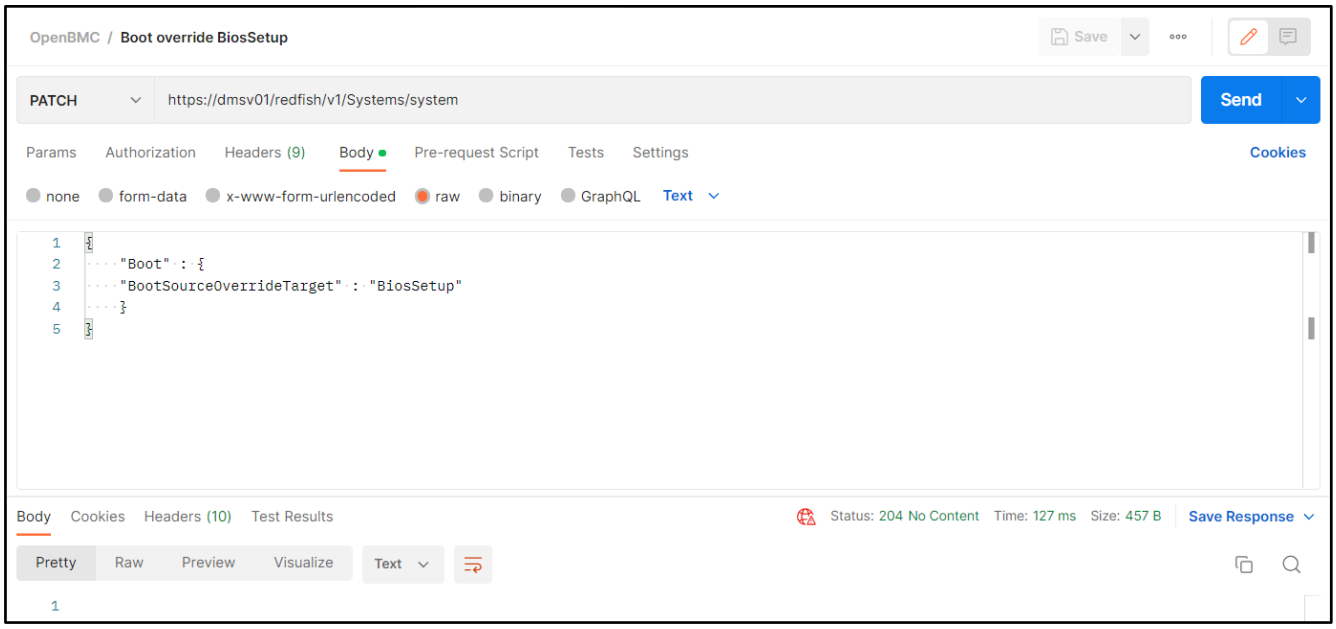


Figure 101: Redfish - Boot override option configured as BiosSetup

3.3.9.4 Force USB Boot Override

Using a PATCH request, it is possible to configure the override function to force the system to boot from the USB.

Function	Boot override - USB
Operation	PATCH
URI	https://<BMC_IP>/redfish/v1/Systems/system

Payload	{
	“Boot”: {
	“BootSourceOverrideTarget” : “Usb”
	}
	}

Header	X-Auth-Token: “<token>”
Expected response	204 No Content
Reply	None

Once the operation is successful, it returns the response “204 No Content” and the host system will boot by means of the USB.

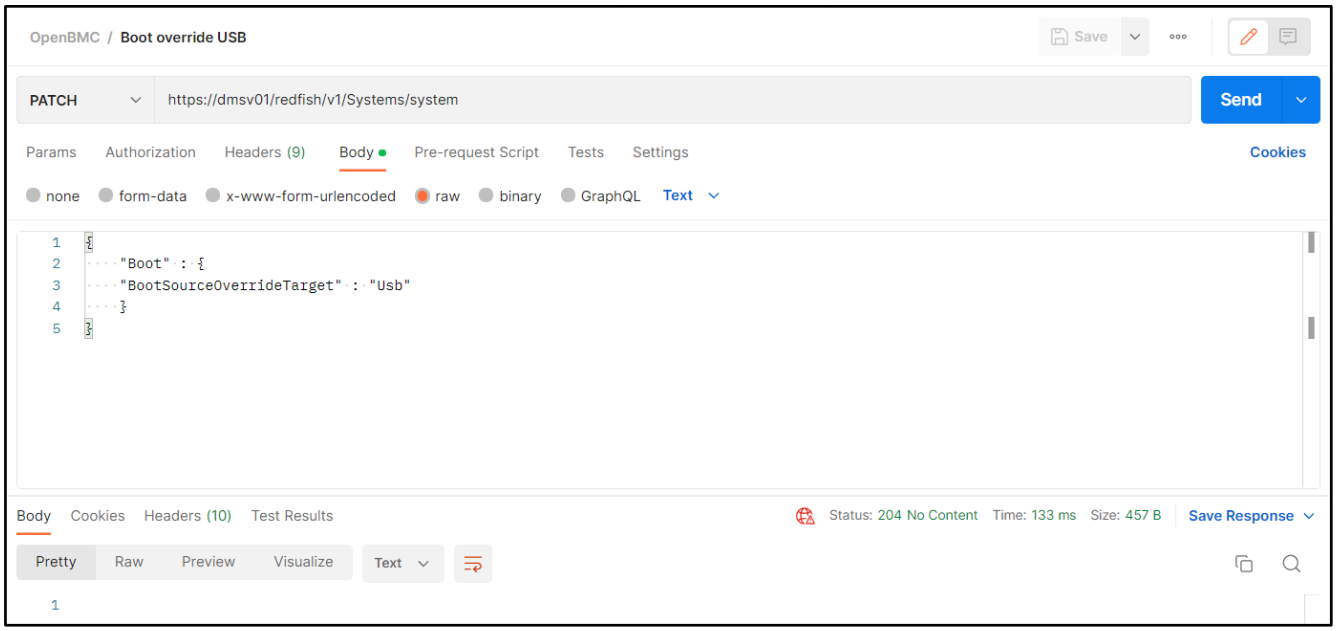


Figure 102: Redfish - Boot override option configured as USB

3.3.9.5 Force HDD Boot Override

Using a PATCH request, it is possible to configure the override function to force the system to boot from the hard disk drive (HDD).

Function	Boot override - HDD
Operation	PATCH
URI	https://<BMC_IP>/redfish/v1/Systems/system

Payload	{
	“Boot”: {
	“BootSourceOverrideTarget” : “Hdd”
	}
	}

Header	X-Auth-Token: “<token>”
Expected response	204 No Content
Reply	None

Once the operation is successful, it returns the response “204 No Content” and the host system will boot by means of the HDD.

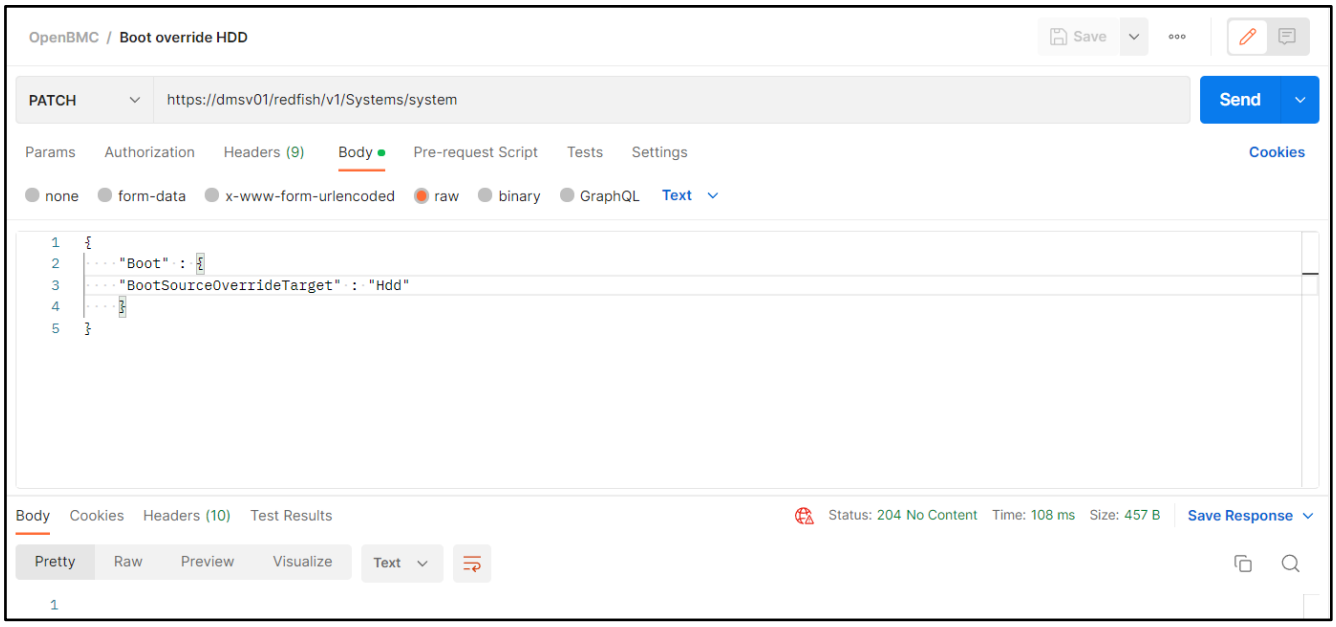


Figure 103: Redfish - Boot override option configured as HDD

3.3.9.6 Disable Boot Override

Using a PATCH request, it is possible to completely disable the boot override function by configuring its target to “None”.

Function	Boot override - Disable
Operation	PATCH
URI	https://<BMC_IP>/redfish/v1/Systems/system

Payload	{
	“Boot”: {
	“BootSourceOverrideTarget” : “None”
	}
	}

Header	X-Auth-Token: “<token>”
Expected response	204 No Content
Reply	None

Once the operation is successful, it returns the response “204 No Content” and the next boot of the host system will be performed using the boot sequence configured in the BIOS menu, without any overriding.

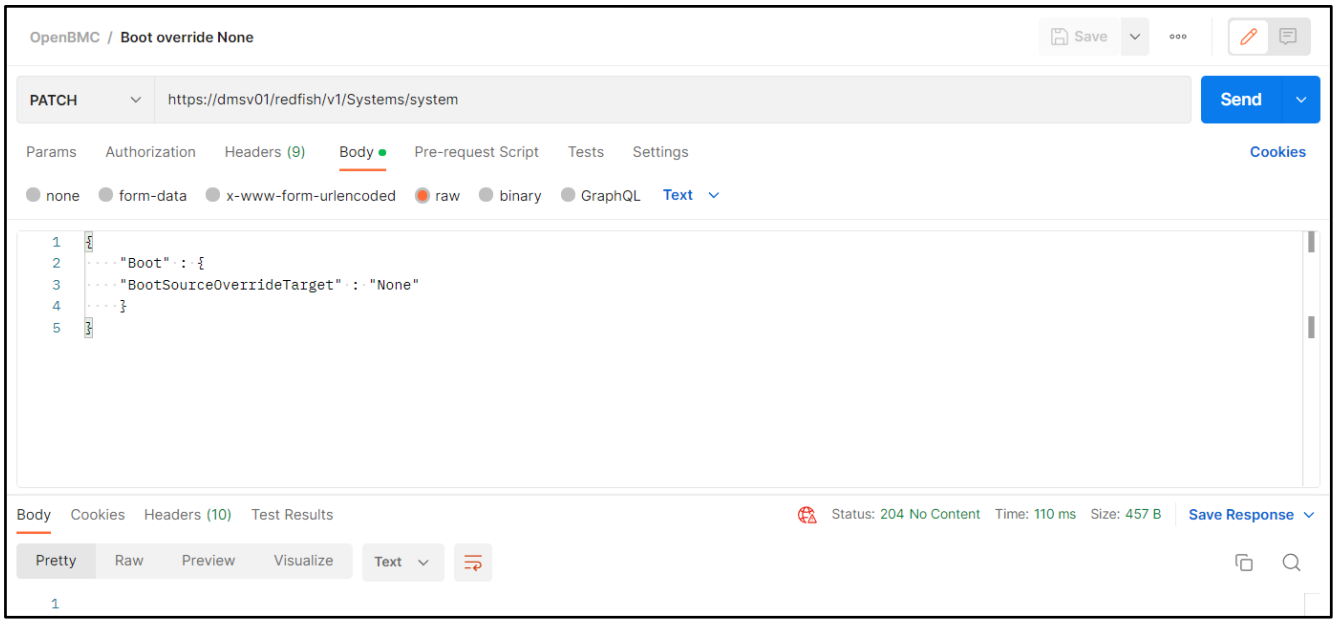


Figure 104: Redfish - Boot override option disabled

3.3.10 LDAP Configuration

The LDAP configuration can be set by means of the redfish. Details regarding the LDAP can be found in section “2.5.1 LDAP”.

3.3.10.1 Open LDAP

Using a PATCH request, it is possible to activate and configure the Open LDAP. The LDAP settings below are sent inside the payload.

- **ServiceEnabled:** true for enabling LDAP or false for disabling it.
- **ServiceAddress (Server URI):** the user must specify the URI (Uniform Resource Identifier) to access the server, starting with the scheme “ldap://”. As an example, a valid entry could be “ldap://mycompany.com”.
- **Username (Bind DN):** the bind DN (Distinguished Name) of the user authenticating to the LDAP directory. As an example, if the username is “user1” and it is part of the “Users” group, the entry should look like this: “CN=user1,OU=Users,DC=mycompany,DC=com”.
- **Password (Bind Password):** the password related to the bind DN above
- **BaseDistinguishedNames (Base DN):** the base DN (Distinguished Name) of the user authenticating to the LDAP directory. An example of a valid entry should look like this: “DC=mycompany,DC=com”.
- **UsernameAttribute (User ID Attribute - optional):** additional user ID attribute (CN), if applicable.
- **GroupsAttribute (Group ID Attribute - optional):** additional group ID attribute (GID), if applicable.

Function	Configure Open LDAP
Operation	PATCH
URI	https://<BMC_IP>/redfish/v1/AccountService

Payload	<pre>{
	  "LDAP": {
	  "Authentication": {
	  "Password": "ldappassword",
	  "Username": "CN=user1,OU=Users,DC=mycompany,DC=com"
	  },
	  "LDAPService": {
	  "SearchSettings": {
	  "BaseDistinguishedNames": [
	  "DC=mycompany,DC=com"
	  ],
	  "GroupsAttribute": "gid",
	  "UsernameAttribute": "cn"
	  }
	  },
	  "ServiceAddresses": [
	  "ldap://mycompany.com"
	  ],
	  "ServiceEnabled": true
	  }
	}</pre>

Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Same as payload

Once the operation is successful, it returns the response “200 OK” and the LDAP is configured according to the data sent inside the payload of the request operation.

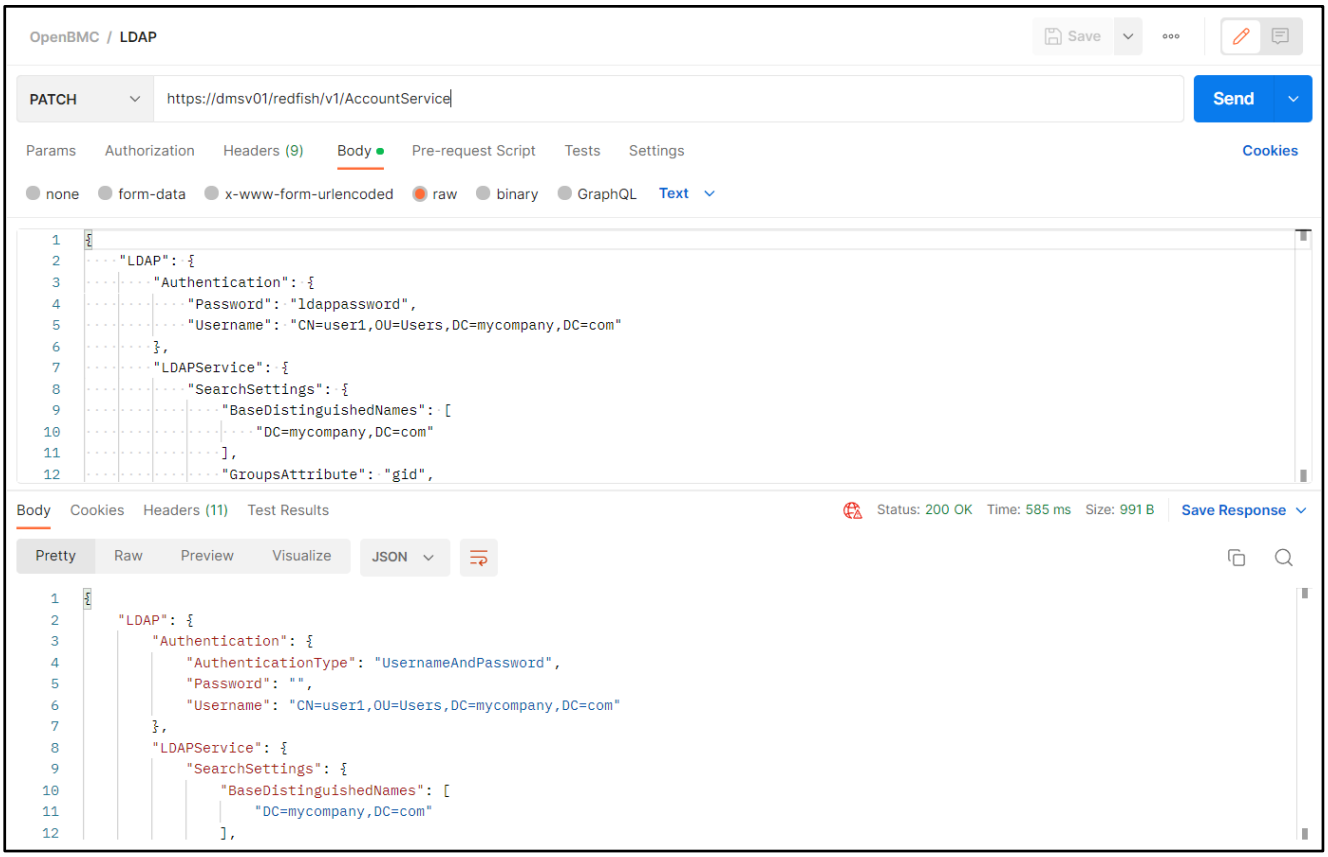


Figure 105: Redfish - Configuring Open LDAP

3.3.10.2 Active Directory

Using a PATCH request, it is possible to activate and configure the Active Directory. The LDAP settings below are sent inside the payload.

- **ServiceEnabled:** true for enabling LDAP or false for disabling it.
- **ServiceAddress (Server URI):** the user must specify the URI (Uniform Resource Identifier) to access the server, starting with the scheme “ldap://”. As an example, a valid entry could be “ldap://mycompany.com”.
- **Username (Bind DN):** the bind DN (Distinguished Name) of the user authenticating to the LDAP directory. As an example, if the username is “user1” and it is part of the “Users” group, the entry should look like this: “CN=user1,OU=Users,DC=mycompany,DC=com”.
- **Password (Bind Password):** the password related to the bind DN above
- **BaseDistinguishedNames (Base DN):** the base DN (Distinguished Name) of the user authenticating to the LDAP directory. An example of a valid entry should look like this: “DC=mycompany,DC=com”.
- **UsernameAttribute (User ID Attribute - optional):** additional user ID attribute (CN), if applicable.
- **GroupsAttribute (Group ID Attribute - optional):** additional group ID attribute (GID), if applicable.

Function	Configure LDAP - Active Directory
Operation	PATCH

URI	https://<BMC_IP>/redfish/v1/AccountService
Payload	{ "ActiveDirectory": { "Authentication": { "Password": "ldappassword", "Username": "CN=user1,OU=Users,DC=mycompany,DC=com" }, "LDAPService": { "SearchSettings": { "BaseDistinguishedNames": ["DC=mycompany,DC=com"], "GroupsAttribute": "gid", "UsernameAttribute": "cn" } }, "ServiceAddresses": ["ldap://mycompany.com"], "ServiceEnabled": true } }
Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Same as payload

Once the operation is successful, it returns the response “200 OK” and the Active Directory is configured according to the data sent inside the payload of the request operation.

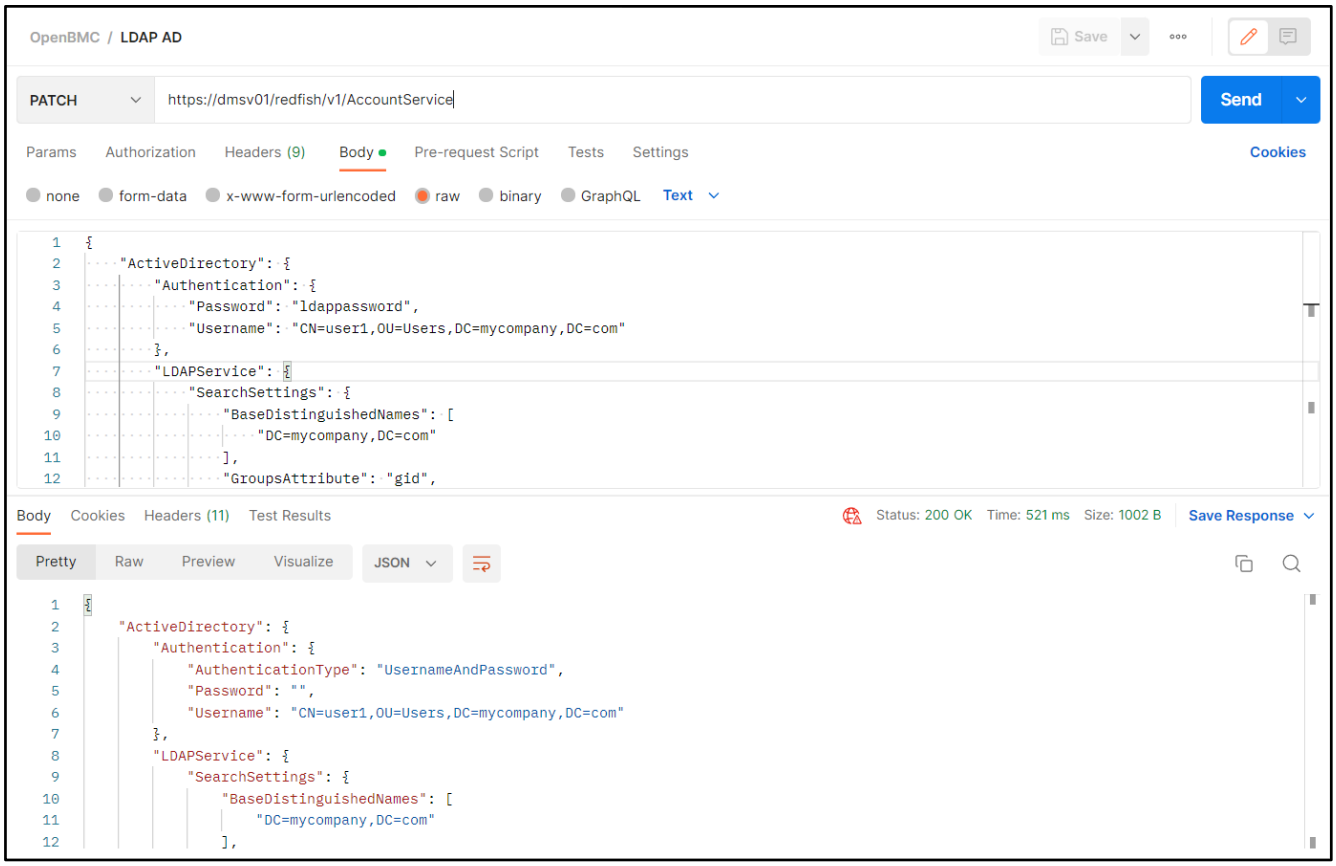


Figure 106: Redfish - Configuring Active Directory

3.3.10.3 Role Groups

Using a PATCH request, it is possible to add or remove role groups. The role group settings below are sent inside the payload.

- **“ActiveDirectory”** or **“LDAP”**: depending on the LDAP service configured, the user must use “ActiveDirectory” for AD or “LDAP” for OpenLDAP. This value is the first string of the payload and the table below is using AD as an example.
- **RemoteGroup (Role Group Name):** the user must specify the name of the role group, as a string.
- **LocalRole (Privilege):** the privilege level for the group of users. There are four options available:
  - Administrator
  - Operator
  - ReadOnly
  - NoAccess

Function	Configure LDAP - Role Groups
Operation	PATCH
URI	https://<BMC_IP>/redfish/v1/AccountService

Payload	{
	"ActiveDirectory": {
	"RemoteRoleMapping": [
	{
	"LocalRole": "Operator",
	"RemoteGroup": "group2"
	}
	]
	}
	}

Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Same as payload

Once the operation is successful, it returns the response “200 OK” and the role group is configured according to the data sent inside the payload of the request operation.

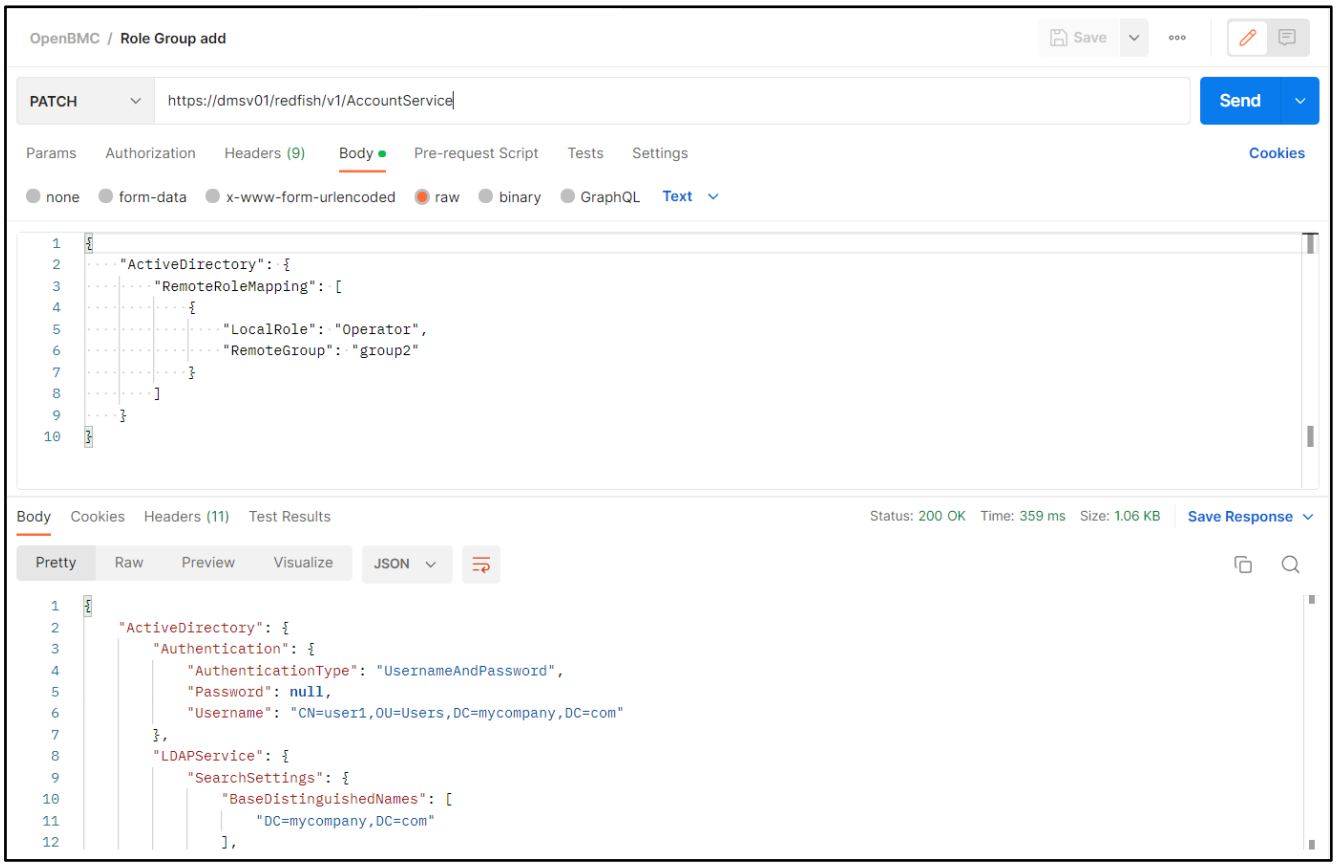


Figure 107: Redfish - Configuring Role Groups

Important: If the user desires to delete all the role groups, it is possible to send a “null” value in the payload, inside the “RemoteRoleMapping” resource, as shown below:

Payload	{
	"ActiveDirectory": {
	"RemoteRoleMapping": [
	null
	]
	}
	}

3.3.11 Users Management

The user's configuration can be managed by means of the redfish. Details regarding the configurations related to users can be found in section “ 2.5.2 Local users”.

3.3.11.1 Change root password

Using a PATCH request, it is possible to change the password of the “root” user.

Important: Please note that the new password must follow the rules defined as per Linux pam_cracklib, which checks the password against dictionary words. Therefore, very simple sequences of characters will not be accepted.

Function	Change root password
Operation	PATCH
URI	https://<BMC_IP>/redfish/v1/AccountService/Accounts/root

Payload	{
	“Password”:”<new password>”
	}

Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	None

Once the operation is successful, it returns the response “200 OK” and the password is changed according to the new string sent inside the payload of the request operation.

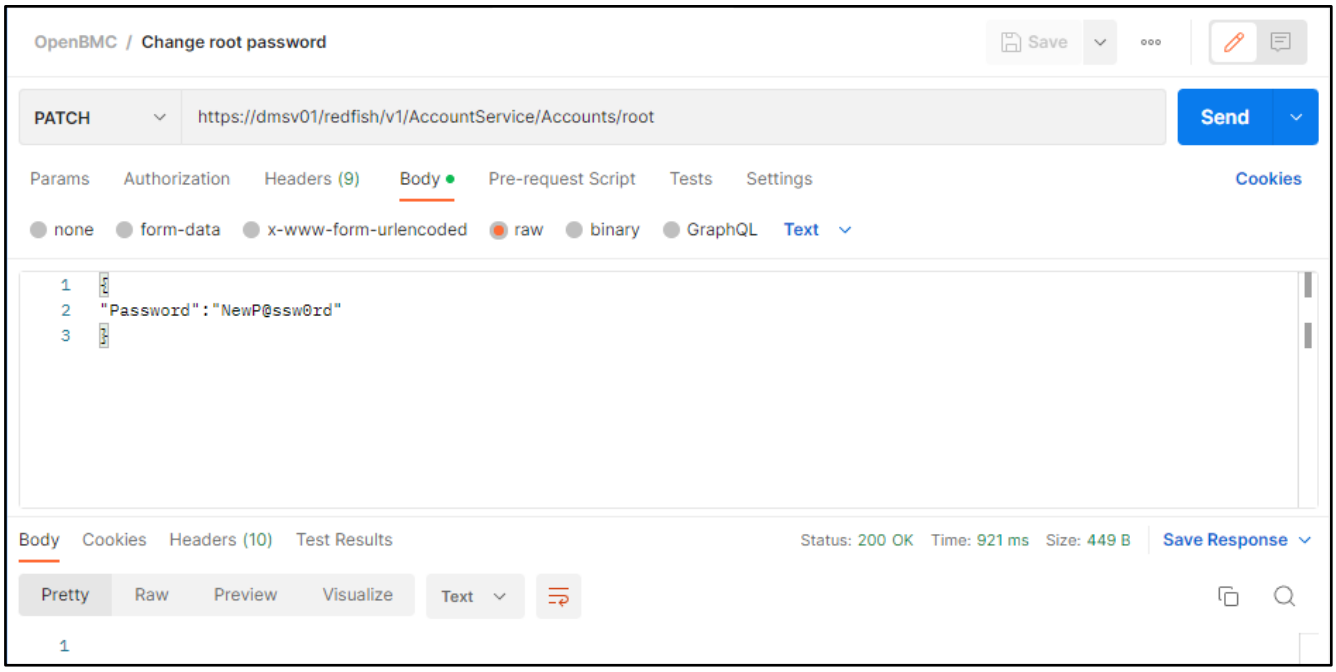


Figure 108: Redfish - Changing the root password

3.3.11.2 Add BMC User

Using a POST request, it is possible to create a new user for the BMC. The following information about the new user is required inside the payload:

- **UserName:** the username cannot start with a number and no special characters are allowed, except the underscore.
- **Password:** the password must have between 8 and 20 characters and must be accepted by the rules defined as per Linux pam_cracklib, which checks the password against dictionary words.
- **Role Id:** defines the privilege level of the user. The available user roles are the following:
  - “Administrator”
  - “Operator”
  - “ReadOnly”
  - “NoAccess”

Function	Add new BMC user
Operation	POST
URI	https://<BMC_IP>/redfish/v1/AccountService/Accounts

Payload	{
	"UserName": "<username>",
	"Password" : "<password>",
	"RoleId" : "<privilege>"
	}

Header	X-Auth-Token: “<token>”
Expected response	201 Created

Reply	{
	"@Message.ExtendedInfo": [
	{
	"@odata.type": "#Message.v1_0_0.Message",
	"Message": "The resource has been created successfully",
	"MessageArgs": [],
	"MessageId": "Base.1.4.0.Created",
	"Resolution": "None",
	"Severity": "OK"
	}
	]
	}

Once the operation is successful, it returns the response “201 Created” and the user is created according to the specifications sent inside the payload of the request.

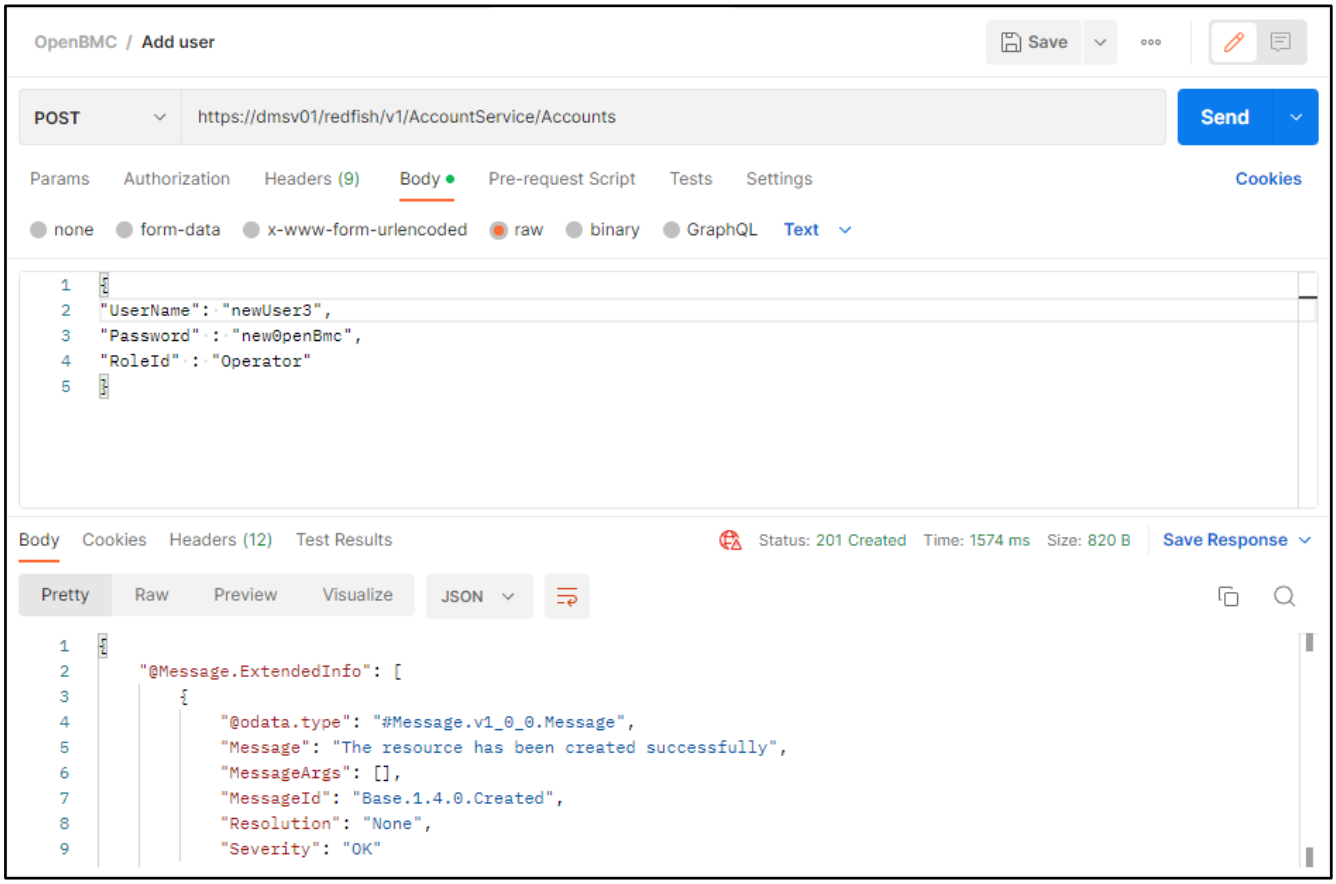


Figure 109: Redfish - Adding a new BMC user

3.3.11.3 Change BMC User Role

Using a PATCH request, it is possible to change the privilege level of a specific BMC User. The username is placed in the URI of the PATCH request and the new privilege to be set is part of the payload. The available user roles are the following:

- “Administrator”
- “Operator”
- “ReadOnly”
- “NoAccess”

Important: it is not possible to change the privilege level of the default “root” user, which is always fixed to “Administrator”.

Function	Change BMC user role
Operation	PATCH
URI	https://<BMC_IP>/redfish/v1/AccountService/Accounts/<username>

Payload	{
	“RoleId”: “<user role>”
	}

Header	X-Auth-Token: “<token>”
Expected response	200 OK

Reply	{
	"@Message.ExtendedInfo": [
	{
	"@odata.type": "#Message.v1_0_0.Message",
	"Message": "Successfully Completed Request",
	"MessageArgs": [],
	"MessageId": "Base.1.4.0.Success",
	"Resolution": "None",
	"Severity": "OK"
	}
	]
	}

Once the operation is successful, it returns the response “200 OK” and the user role is changed according to the string sent inside the payload of the request.

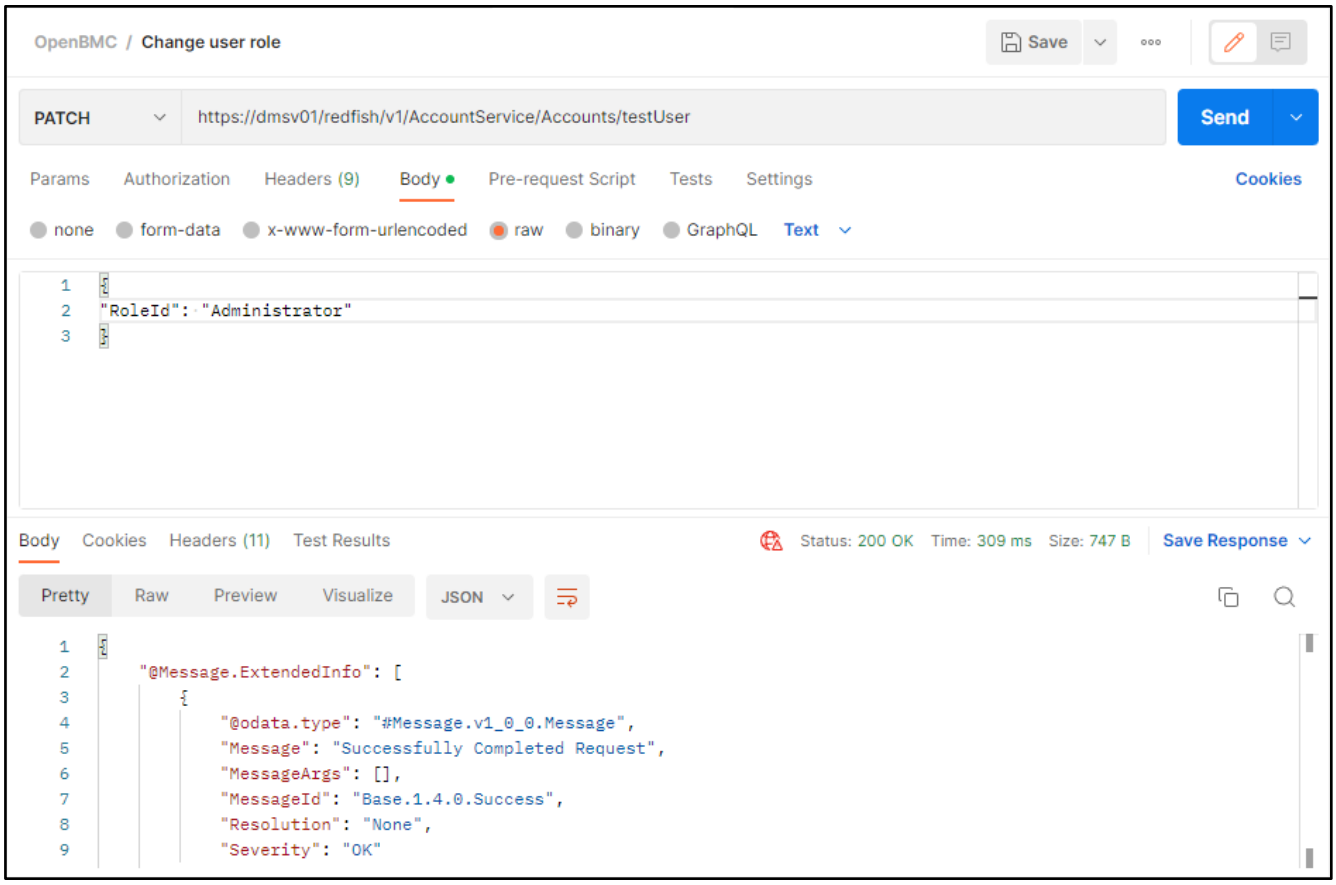


Figure 110: Redfish - Changing the user privilege

3.3.11.4 Change BMC User Password

Using a PATCH request, it is possible to change the password of any user. The username is placed in the URI of the PATCH request and the new password to be set is part of the payload.

Important: Please note that the new password must follow the rules defined as per Linux pam_cracklib, which checks the password against dictionary words. Therefore, very simple sequences of characters will not be accepted.

Function	Change BMC user password
Operation	PATCH
URI	https://<BMC_IP>/redfish/v1/AccountService/Accounts/<username>

Payload	{
	“Password”:”<new password>”
	}

Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	None

Once the operation is successful, it returns the response “200 OK” and the password is changed according to the new string sent inside the payload of the request.

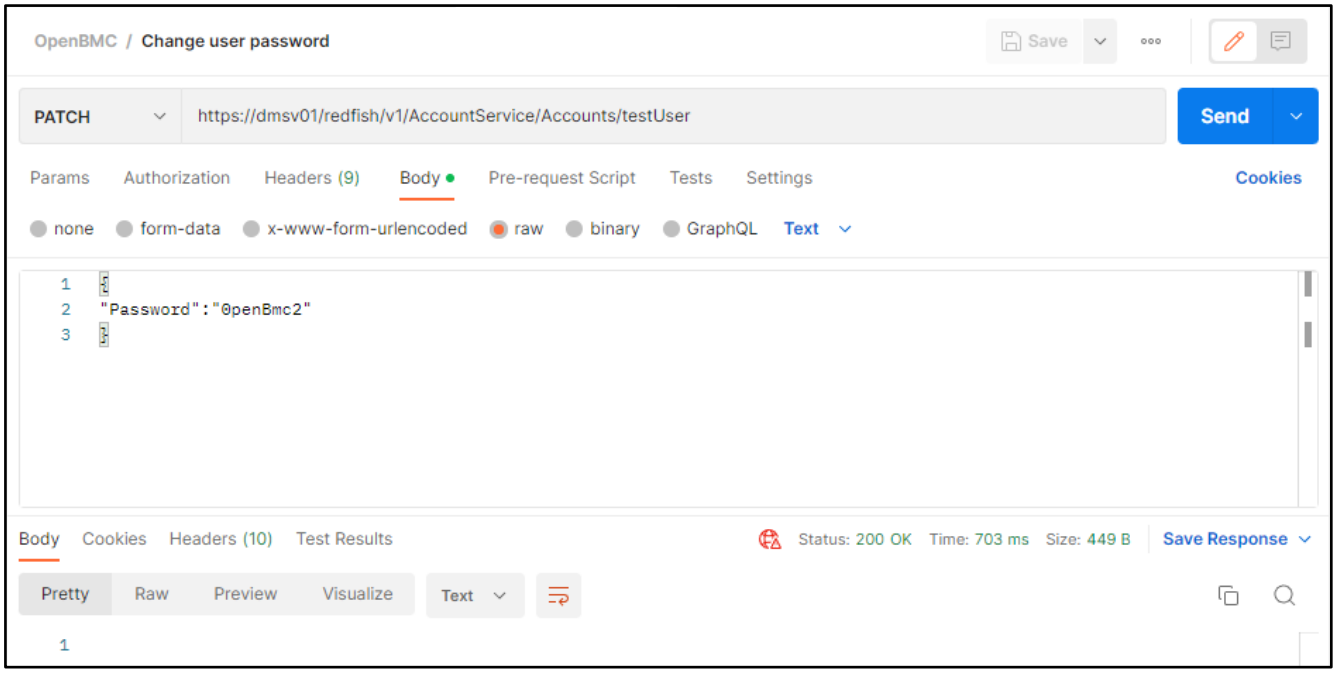


Figure 111: Redfish - Changing an user password

3.3.11.5 Delete BMC User

Using a DELETE request, it is possible to delete an user. The username is placed in the URI of the DELETE request

Function	Delete BMC user
Operation	DELETE
URI	https://<BMC_IP>/redfish/v1/AccountService/Accounts/<username>
Payload	None
Header	X-Auth-Token: “<token>”
Expected response	200 OK

Reply	{
	"@Message.ExtendedInfo": [
	{
	"@odata.type": "#Message.v1_0_0.Message",
	"Message": "The account was successfully removed.",
	"MessageArgs": [],
	"MessageId": "Base.1.4.0.AccountRemoved",
	"Resolution": "No resolution is required.",
	"Severity": "OK"
	}
	]
	}

Once the operation is successful, it returns the response “200 OK” and the user is deleted.

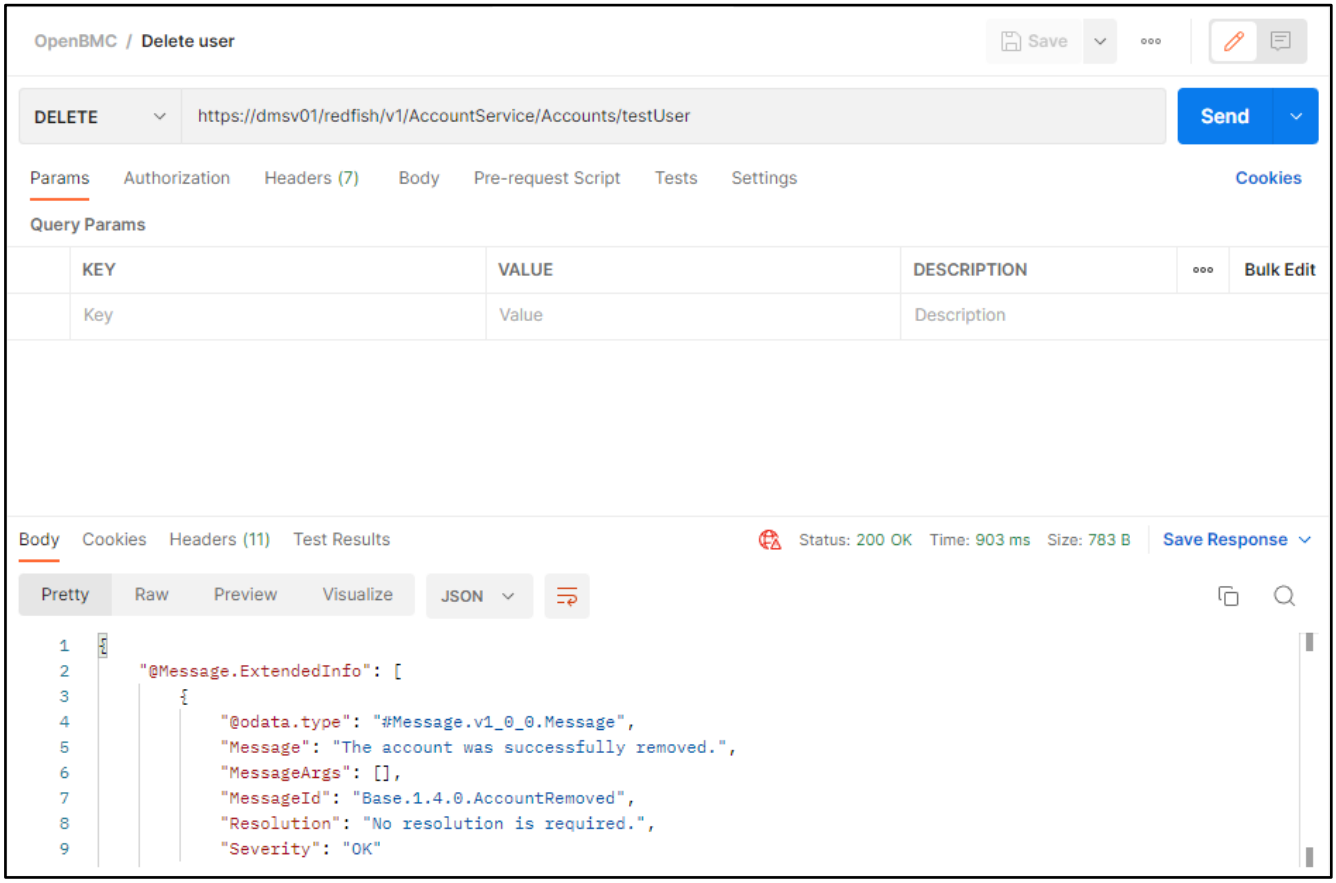


Figure 112: Redfish - Deleting an user

3.3.12 FW Update

The user can update the BMC FW or the BIOS FW by means of the redfish. Details regarding FW update can be found in section “2.4.2.2 FW update process - BMC or BIOS”.

3.3.12.1 Update BMC Firmware

Using a POST request, it is possible to upload an image file for updating the BMC FW. The new FW image file must be sent within the POST request. Using Postman, for example, the file is selected by means of the “binary” option inside the “Body” tab.

Function	Update BMC FW
Operation	POST
URI	https://<BMC_IP>/redfish/v1/UpdateService
Payload	Upload file
Header	X-Auth-Token: “<token>”
Expected response	200 OK

Reply	{
	"@odata.id": "/redfish/v1/TaskService/Tasks/0",
	"@odata.type": "#Task.v1_4_3.Task",
	"Id": "0",
	"TaskState": "Running",
	"TaskStatus": "OK"
	}

Once the operation is successful, it returns the response “200 OK” and the FW is uploaded and ready for the update.

Important: the POST request uploads the FW image and prepares it for the update. However, the FW will only be updated after a manual reboot of the BMC, which can be performed by following the procedure described in section “3.3.14.1 Reboot BMC”.

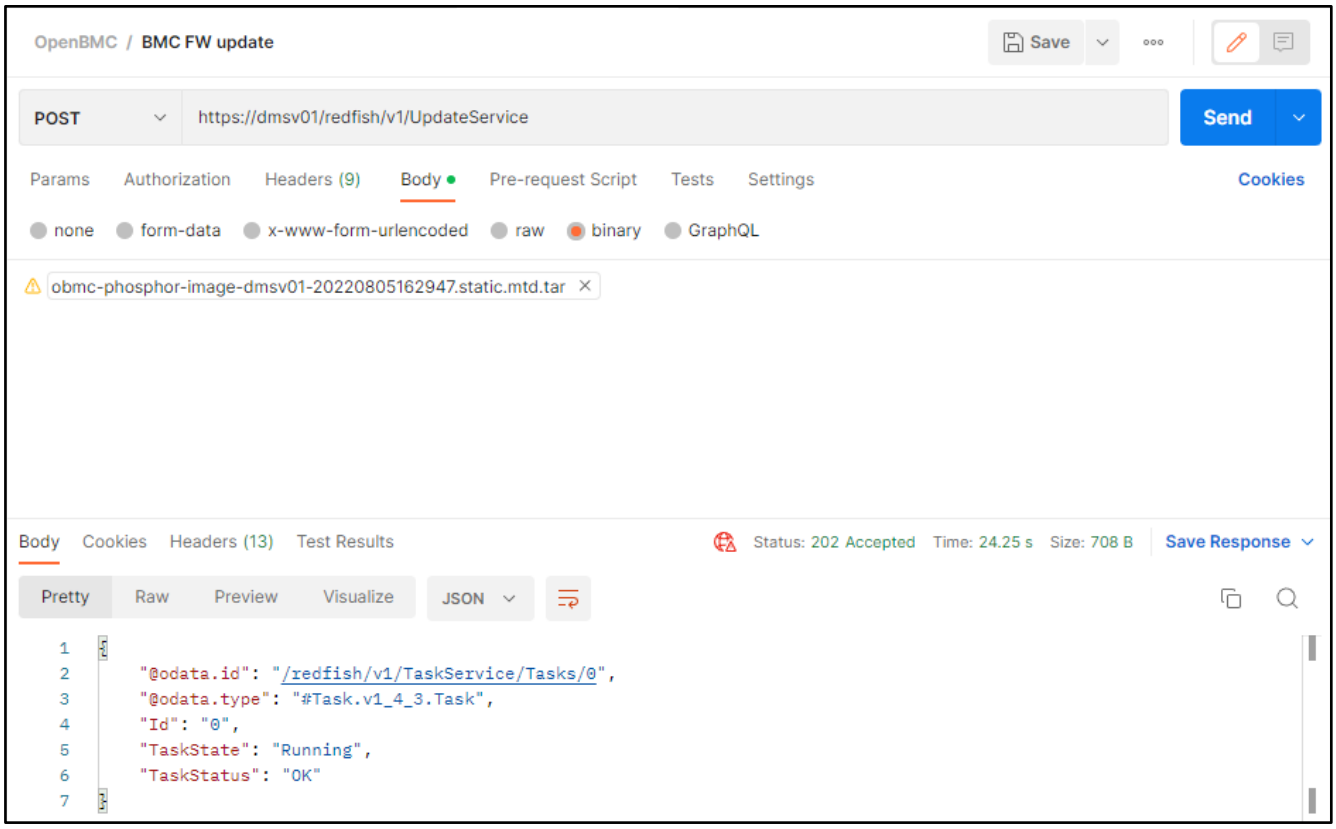


Figure 113: Redfish - BMC FW update

3.3.12.2 Update BIOS Firmware

Using a POST request, it is possible to upload an image file for updating the BIOS FW. The new FW image file must be sent within the POST request. Using Postman, for example, the file is selected by means of the “binary” option inside the “Body” tab.

Function	Update BIOS FW
Operation	POST
URI	https://<BMC_IP>/redfish/v1/UpdateService
Payload	Upload file
Header	X-Auth-Token: “<token>”
Expected response	200 OK

Reply	{
	"@odata.id": "/redfish/v1/TaskService/Tasks/0",
	"@odata.type": "#Task.v1_4_3.Task",
	"Id": "0",
	"TaskState": "Running",
	"TaskStatus": "OK"
	}

Once the operation is successful, it returns the response “200 OK” and the FW is uploaded and ready for the update.

Important: the POST request uploads the image and automatically updates the BIOS FW. If the host processors are turned on during the execution of the POST request, the system will force a host shutdown in order to perform the update. So, it is recommended to manually power the host processors off before running this request, in order to avoid the forced shutdown.

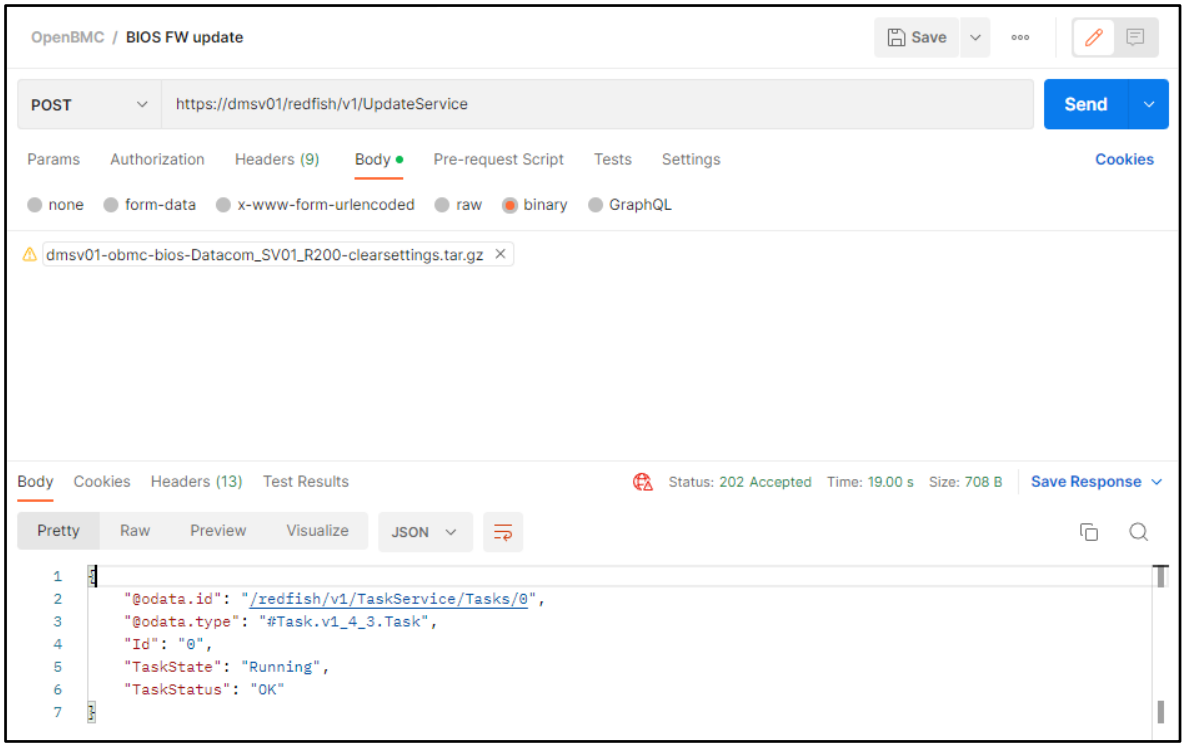


Figure 114: Redfish - BIOS FW update

3.3.13 Logging

The user can view and delete system logs by means of the redfish. Details regarding the system logs can be found in section “2.2.1 Event log”.

3.3.13.1 View Log Entries

Using a GET request, it is possible to retrieve all the log entries.

Function	View log entries
Operation	GET
URI	https://<BMC_IP>/redfish/v1/Systems/system/LogServices/EventLog/Entries
Payload	None
Header	X-Auth-Token: “<token>”
Expected response	200 OK

Reply	Please, see the example below.

As an example, the excerpt below shows the data provided by the DM-SV01 mainboard using the redfish GET request. The user can check relevant information of each event logged in the BMC.


```
{
   "@odata.id": "/redfish/v1/Systems/system/LogServices/EventLog/Entries",
   "@odata.type": "#LogEntryCollection.LogEntryCollection",
   "Description": "Collection of System Event Log Entries",
   "Members": [
      {
         "@odata.id": "/redfish/v1/Systems/system/LogServices/EventLog/Entries/70",
         "@odata.type": "#LogEntry.v1_4_0.LogEntry",
         "Created": "1970-01-01T00:01:10+00:00",
         "EntryType": "Event",
         "Id": "70",
         "Message": "FAN_LEFT sensor crossed a warning low threshold going low. Reading=0.000000 Threshold=1500.000000.",
         "MessageArgs": [
            "FAN_LEFT",
            "0.000000",
            "1500.000000"
         ],
         "MessageId": "OpenBMC.0.1.SensorThresholdWarningLowGoingLow",
         "Name": "System Event Log Entry",
         "Severity": "Warning"
      },
      {
         "@odata.id": "/redfish/v1/Systems/system/LogServices/EventLog/Entries/74",
         "@odata.type": "#LogEntry.v1_4_0.LogEntry",
         "Created": "1970-01-01T00:01:14+00:00",
         "EntryType": "Event",
         "Id": "74",
         "Message": "FAN_LEFT sensor crossed a critical low threshold going low. Reading=0.000000 Threshold=1000.000000.",
         "MessageArgs": [
            "FAN_LEFT",
            "0.000000",
            "1000.000000"
         ],
         "MessageId": "OpenBMC.0.1.SensorThresholdCriticalLowGoingLow",
         "Name": "System Event Log Entry",
```

```
            "Severity": "Critical"
        },
        {
            "@odata.id": "/redfish/v1/Systems/system/LogServices/EventLog/Entries/82",
            "@odata.type": "#LogEntry.v1_4_0.LogEntry",
            "Created": "1970-01-01T00:01:22+00:00",
            "EntryType": "Event",
            "Id": "82",
            "Message": "FAN_RIGHT sensor crossed a warning low threshold going low. Reading=0.000000 Threshold=1500.000000.",
            "MessageArgs": [
                "FAN_RIGHT",
                "0.000000",
                "1500.000000"
            ],
            "MessageId": "OpenBMC.0.1.SensorThresholdWarningLowGoingLow",
            "Name": "System Event Log Entry",
            "Severity": "Warning"
        },
        {
            "@odata.id": "/redfish/v1/Systems/system/LogServices/EventLog/Entries/90",
            "@odata.type": "#LogEntry.v1_4_0.LogEntry",
            "Created": "1970-01-01T00:01:30+00:00",
            "EntryType": "Event",
            "Id": "90",
            "Message": "FAN_RIGHT sensor crossed a critical low threshold going low. Reading=0.000000 Threshold=1000.000000.",
            "MessageArgs": [
                "FAN_RIGHT",
                "0.000000",
                "1000.000000"
            ],
            "MessageId": "OpenBMC.0.1.SensorThresholdCriticalLowGoingLow",
            "Name": "System Event Log Entry",
            "Severity": "Critical"
        }
    ],
    "Members@odata.count": 4,
    "Name": "System Event Log Entries"
}
```


Once the operation is successful, it returns the response “200 OK” and the logs are shown in the body of the request response.

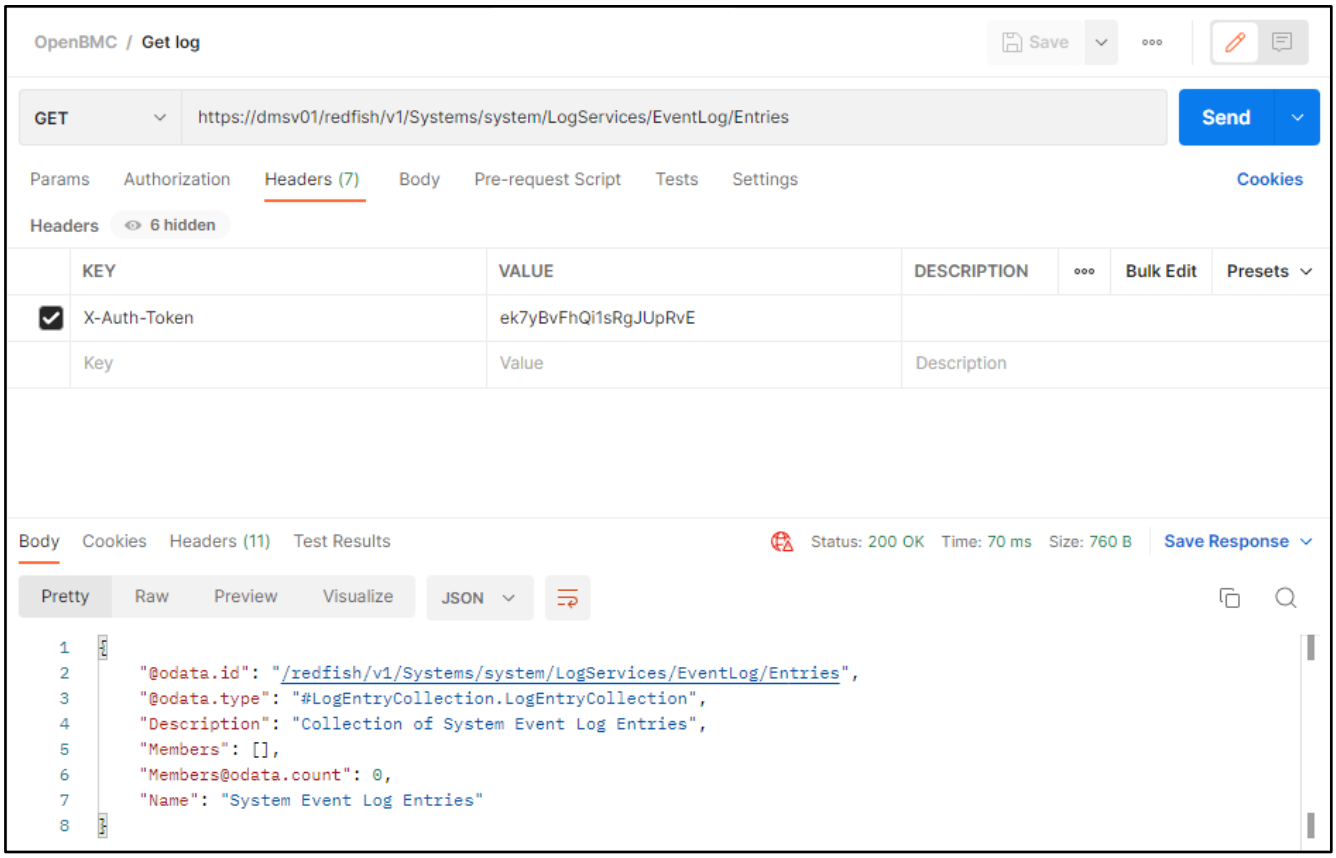


Figure 115: Redfish - System log retrieve

3.3.13.2 Delete Log Entries

Using a POST request, it is possible to clear all the log entries.

Important: please be careful because this operation cannot be reverted. Once the logs are deleted, their information is completely lost.

Function	Clear log entries
Operation	POST
URI	https://<BMC_IP>/redfish/v1/Systems/system/LogServices/EventLog/Actions/LogService.ClearLog
Payload	None
Header	X-Auth-Token: “<token>”
Expected response	200 OK

Reply	{
	"@Message.ExtendedInfo": [
	{
	"@odata.type": "#Message.v1_0_0.Message",
	"Message": "Successfully Completed Request",
	"MessageArgs": [],
	"MessageId": "Base.1.4.0.Success",
	"Resolution": "None",
	"Severity": "OK"
	}
	]
	}

Once the operation is successful, it returns the response “200 OK” and the logs are completely deleted.

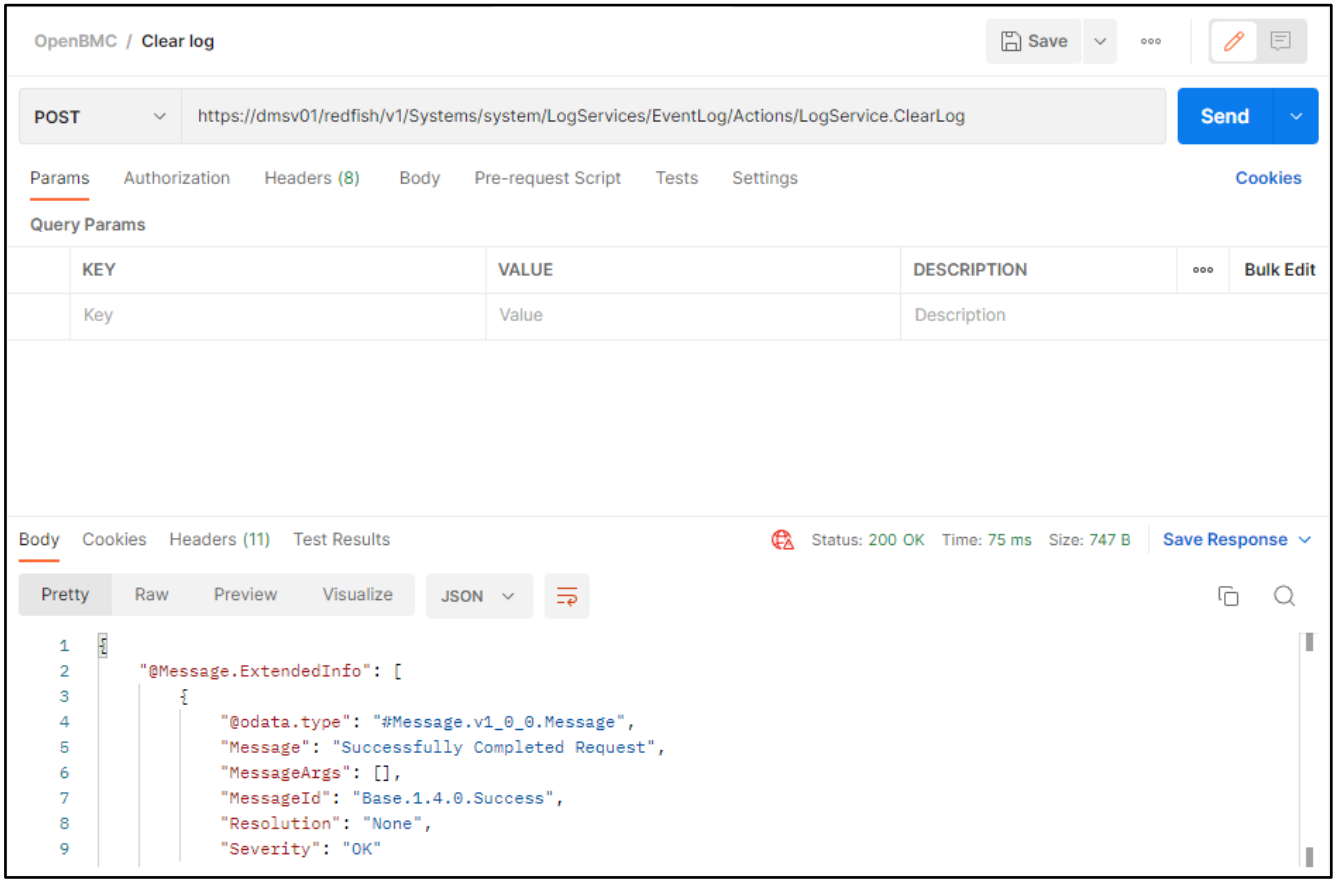


Figure 116: Redfish - System log delete

3.3.14 BMC Reset

The user can perform a reboot in the BMC or reset it to factory defaults by means of the redfish. Details regarding the BMC reboot can be found in section “2.3.3 Reboot BMC” and information about the BMC reset to factory defaults can be found in section “2.4.2.3 Factory Reset - BIOS and BMC”.

3.3.14.1 Reboot BMC

Using a POST request, it is possible to reboot the BMC. As explained in section “2.3.3 Reboot BMC”, the host system is not affected by the BMC reboot and the BMC will become inaccessible during the reboot process.

Function	BMC reboot
Operation	POST
URI	https://<BMC_IP>/redfish/v1/Managers/bmc/Actions/Manager.Reset

Payload	{
	“ResetType”:”GracefulRestart”
	}

Header	X-Auth-Token: “<token>”
Expected response	200 OK

Reply	{
	"@Message.ExtendedInfo": [
	{
	"@odata.type": "#Message.v1_0_0.Message",
	"Message": "Successfully Completed Request",
	"MessageArgs": [],
	"MessageId": "Base.1.4.0.Success",
	"Resolution": "None",
	"Severity": "OK"
	}
	]
	}

Once the operation is successful, it returns the response “200 OK” and the BMC is immediately rebooted.

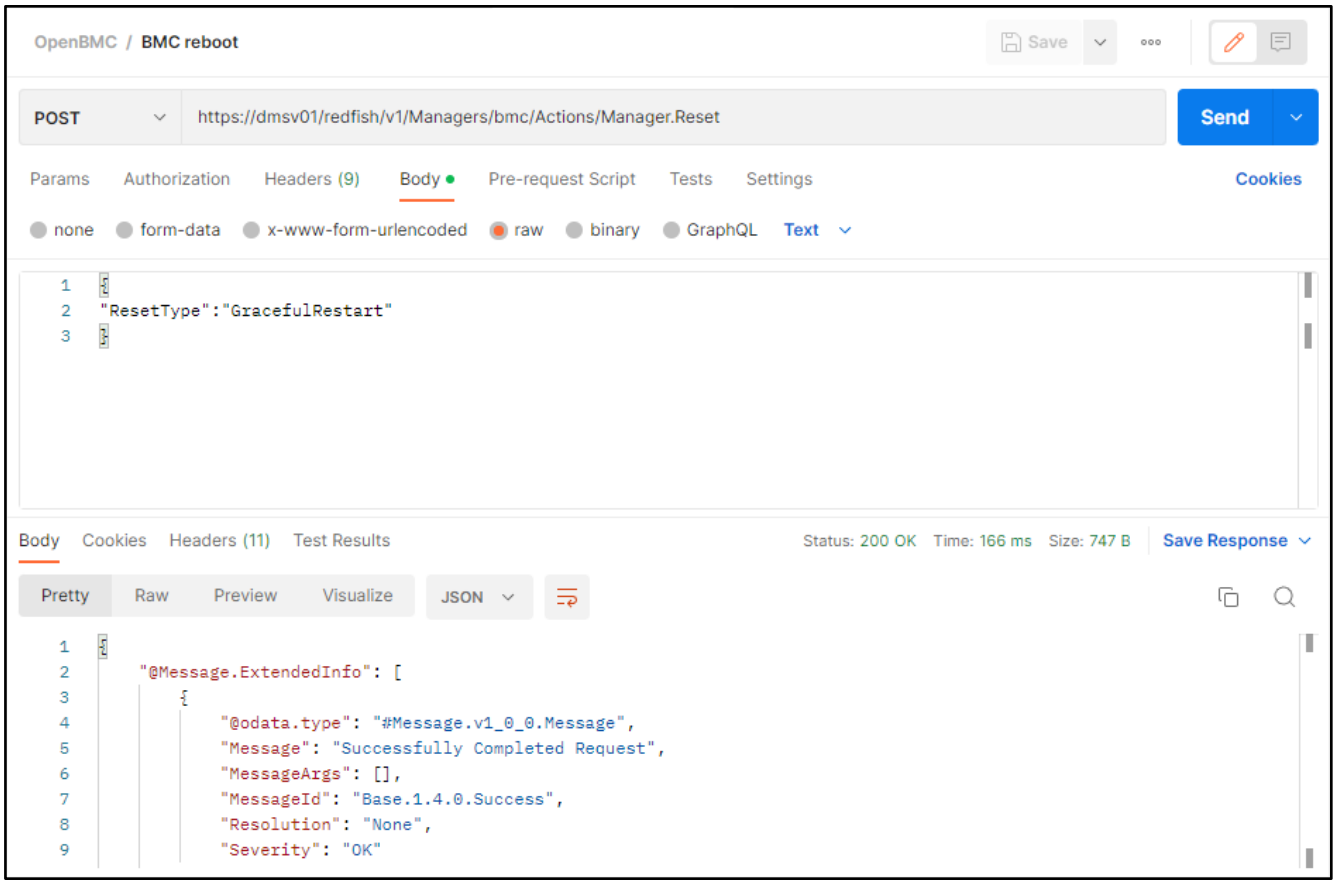


Figure 117: Redfish - BMC reboot

3.3.14.2 Reset BMC to Factory Defaults

Using a POST request, it is possible to reset the BMC to factory defaults.

Function	BMC factory reset
Operation	POST
URI	https://<BMC_IP>/redfish/v1/Managers/bmc/Actions/Manager.ResetToDefaults

Payload	{
	“ResetToDefaultsType”:”ResetAll”
	}

Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	None

Once the operation is successful, it returns the response “200 OK” and all the BMC settings are reverted back to the factory defaults, including network settings, users, root password, etc.

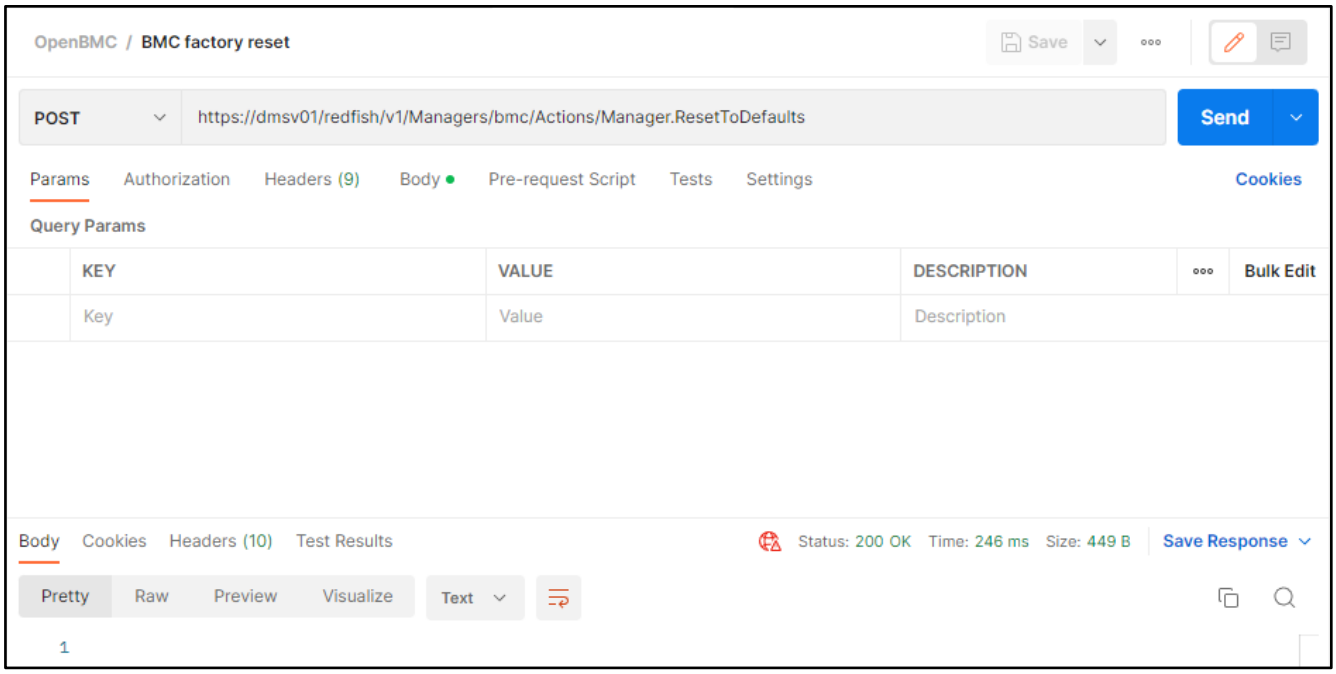


Figure 118: Redfish - BMC factory reset

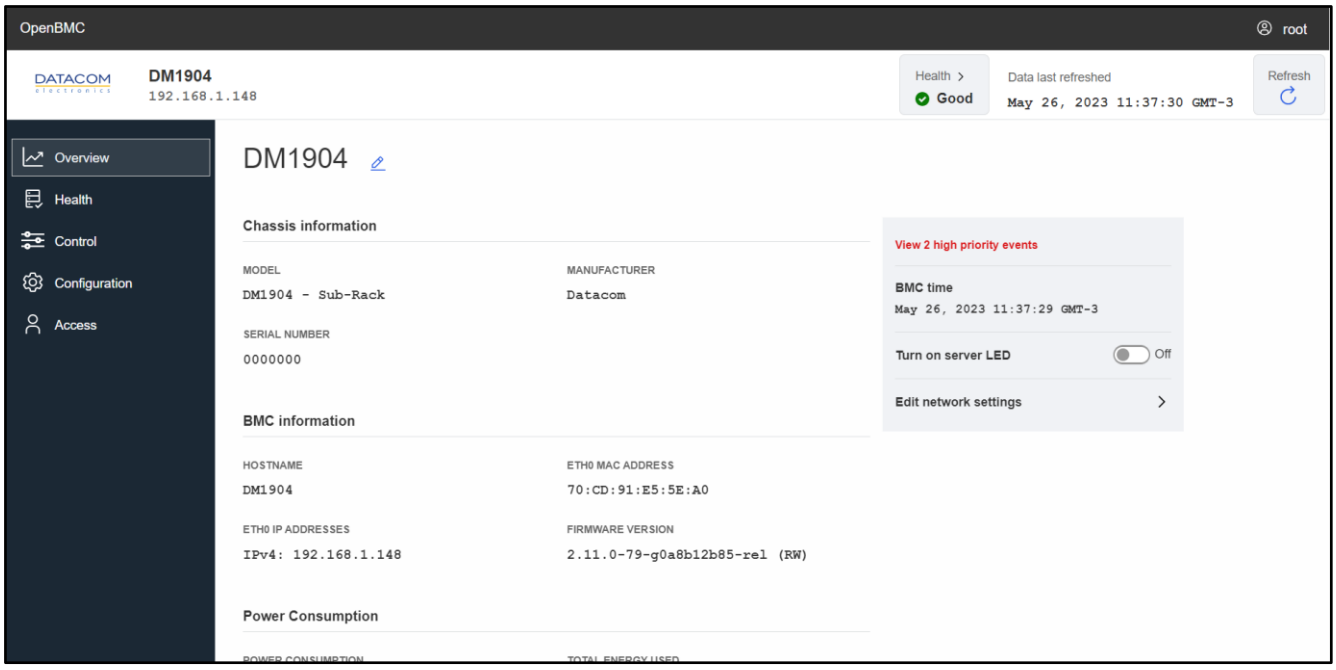
4 Supervisory Board BMC

The Supervisory Board is a Datacom proprietary module used in the DM1904 sub-rack for managing the power supplies connected to the chassis. Details regarding the Supervisory Board module and its functionalities can be found at “DM-SV01 - Product Manual” (1).

Most of the graphical user interface of the Supervisory Board is very similar to the DM-SV01 server BMC. This section provides an overview of each menu and highlights the differences between the Supervisory Board BMC and the DM-SV01 BMC. In cases where the similarity is very high, a link to the corresponding sub chapter of section 3 is suggested.

4.1 Overview Menu

The overview menu is very similar to the corresponding screen at the DM-SV01 server BMC. The differences are restricted to the information displayed in the screen, which are adjusted to match the Supervisory Board inventory data.



The following information is shown at the “Overview” menu:

1. **Hostname.**
2. **Chassis information:** shows general information about the DM1904 chassis, such as:
    a. Model.
    b. Manufacturer.
    c. Serial number.
3. **BMC information:** shows general information about the BMC, such as:
    a. Hostname.
    b. Mac addresses of the network interface eth0.
    c. IP addresses of the network interface eth0.
    d. BMC firmware version.

4. **Power Consumption:** displays the power consumption of the chassis in Watts and the total energy used in kWh (the value shown corresponds to the power consumption when the overview page was last loaded).
5. **High priority events:** displays high priority events if available.

The shortcuts below are available in the overview menu:

1. **Edit network settings:** redirects the user to the “Network settings” menu. Details regarding this function can be found in section “4.4.1 Network Settings”.

Additionally, there is a button in the header of the web page:

- **Server Health:** provides information about the health state of the Supervisory Board, indicating if there is some critical alarm or not. Details of this function can be found in section “4.2.1 Event Log”. The state of the server health button is updated only after the web page is refreshed.

4.2 Health Menu

4.2.1 Event Log

The “Event Log” menu of the Supervisory Board BMC is identical to the corresponding menu from the DM-SV01 server BMC. Therefore, refer to section “2.2.1 Event log” for details regarding this functionality.

4.2.2 Hardware Status

The “Hardware status” menu is used to check the inventory information of the following chassis components:

- Supervisory Board
- DM1904 Sub-rack
- DM1904 Power supplies

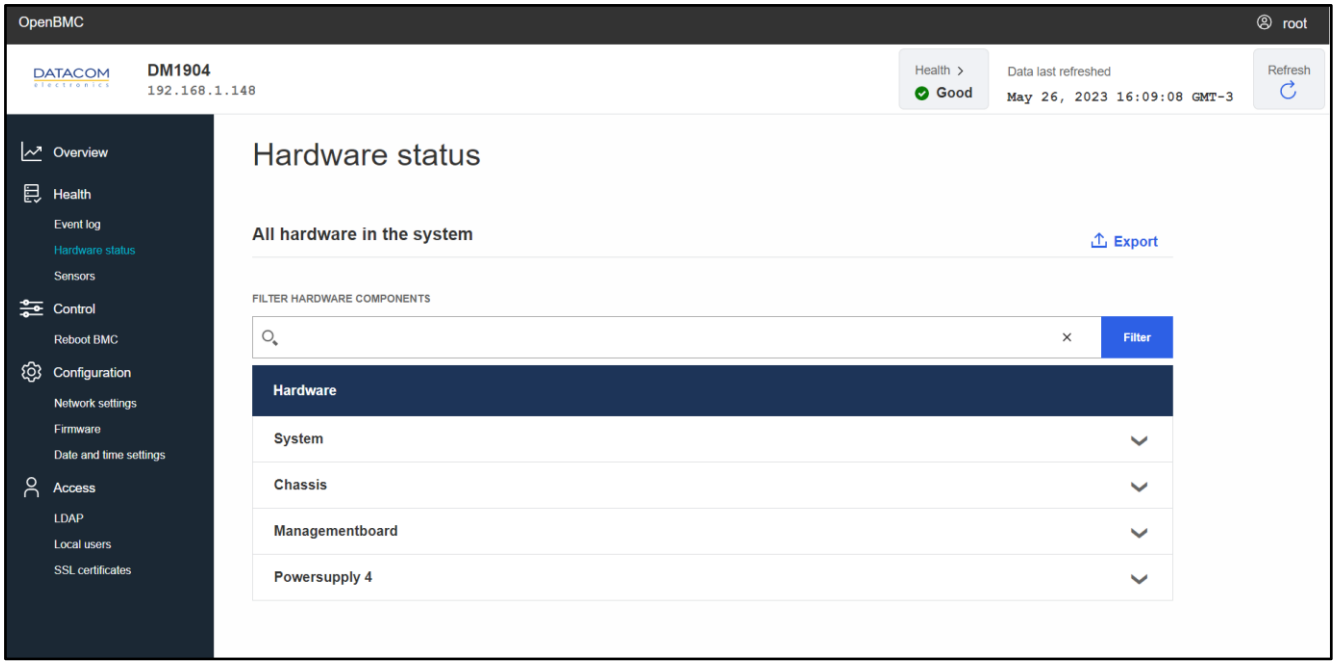


Figure 119: Supervisory Board hardware status menu

The “System” and “Chassis” fields provide inventory information about the DM1904 chassis. The “Managementboard”, on the other hand, presents the inventory details related to the Supervisory Board itself.

Additionally, the Supervisory Board reads the inventory data from all the power supplies installed in the DM1904 sub-rack (slots 1 up to 4) and provides this information to the user in the fields “Powersupply <slot>”. An example of the inventory data from a DM1904 power supply can be seen in Figure 120.

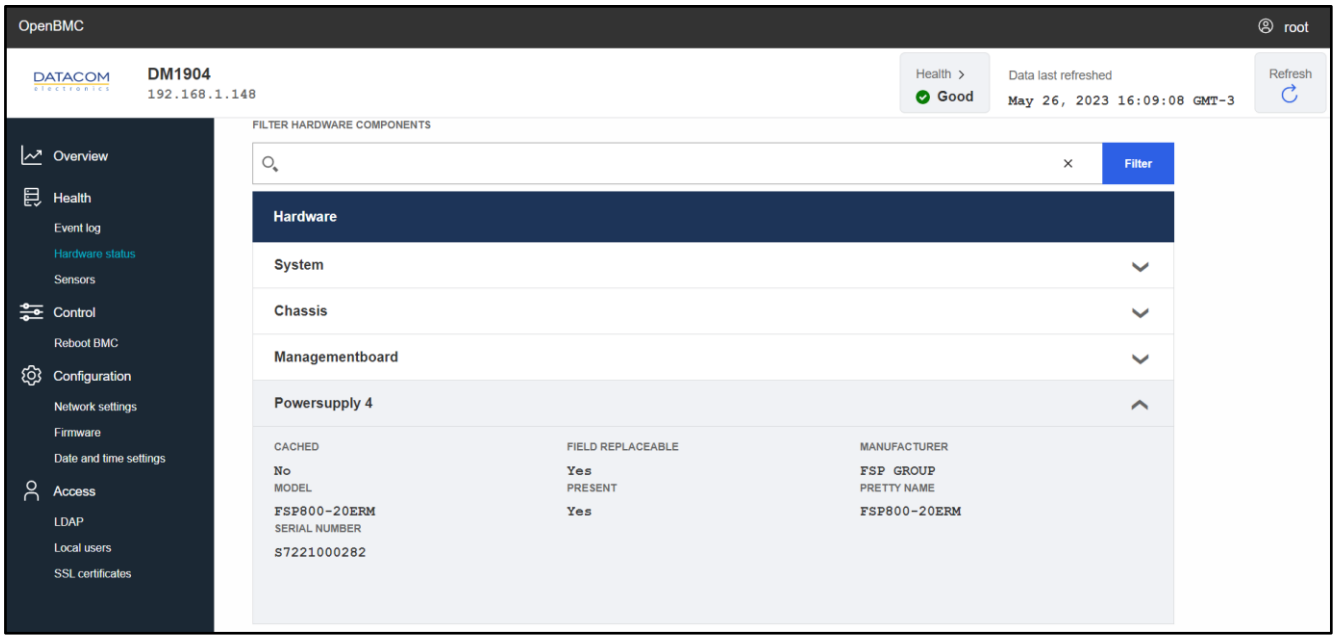


Figure 120: Hardware status menu - DM1904 power supply

4.2.3 Sensors

The Supervisory Board has a set of sensors which are responsible for monitoring the voltage, current, temperature and power consumption of the power supplies plugged in the DM1904 chassis.

The Figure 121 below shows the “Sensors” screen available at the BMC web management interface of the Supervisory Board.

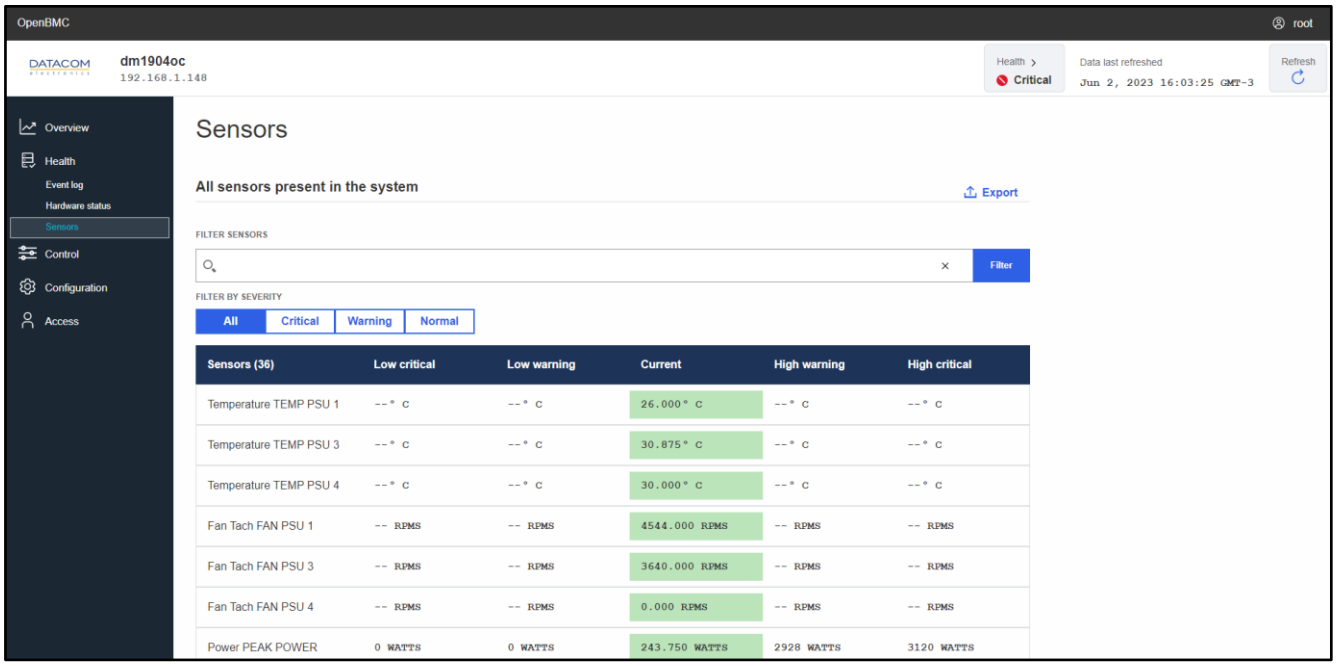


Figure 121: Supervisory Board BMC Sensors menu

The table below shows the description of each sensor and its respective alias in the BMC web management interface.

Sensor name in the BMC web GUI	Sensor description
Temperature TEMP PSU <PSU_number>	Temperature of the respective PSU.
Fan Tach FAN PSU <PSU_number>	Fan speed measured in the tachometer of the respective PSU.
Power PEAK POWER	Maximum instantaneous power value consumed by the whole DM1904 chassis.
Power PIN PSU <PSU_number>	Input power of the respective PSU.
Power POUT PSU <PSU_number>	Output power of the respective PSU.
Power Total Power	DM1904 chassis total instantaneous input power.
Voltage VDD 5 DUAL	Supervisory Board internal 5V power supply.
Voltage VDD 12V	Supervisory Board 12V power input.
Voltage VDD 33 DUAL	Supervisory Board internal 3.3V power supply.
Voltage VDD CORE DUAL	Supervisory Board internal 1.15V BMC core voltage.
Voltage VDD MEM DUAL	Supervisory Board internal 1.25V BMC DDR4 voltage.
Voltage VDD VPP DUAL	Supervisory Board internal 2.50V BMC DDR4 voltage.
Voltage VIN PSU <PSU_number>	Input voltage of the respective PSU.

Voltage VOUT PSU <PSU_number>	Output voltage of the respective PSU.
Current IIN PSU <PSU_number>	Input current of the respective PSU.
Current IOUT PSU <PSU_number>	Output current of the respective PSU.
Energy TOTAL ENERGY	Total energy consumed cumulatively by the system, in “joules”. The joule (symbol: J) is the unit of energy in the International System of Units (SI). 1 J = 1 W.s. Or 1kWh = 3.600.000 J This sensor can be manually reset to zero by the user whenever necessary (refer to section 4.2.3.1.1 Power sensors Reset).

Table 10: Supervisory Board sensors description

4.2.3.1 Power consumption sensors

The DM1904 chassis has some sensors that are capable of monitoring the power consumption of the system. The following power sensors are available:

- **Total Energy:** this sensor displays the accumulated energy consumed by the whole DM1904 chassis in “joules”. The measurement is cumulative and it comprises the whole energy consumed by the system since the server has been turned on until the current moment. The user can reset the total energy counter at any time, in order to monitor the energy consumption in the desired time period. For details regarding the reset of the sensor, please refer to section “4.2.3.1.1 Power sensors Reset”.
- **Peak Power:** this sensor measures the maximum instantaneous power consumed by the DM1904 chassis since it has been turned on. The value of this sensor will be updated only if the BMC measures a peak power value higher than the current value being shown in the sensor. The peak power sensor can also be reset by the user whenever necessary. For details regarding the reset of the sensor, please refer to section “4.2.3.1.1 Power sensors Reset”.
- **Total Power:** this sensor shows the instantaneous power being consumed by the DM1904 chassis. In the BMC web GUI, the value of the sensor is updated only when the web page is refreshed.

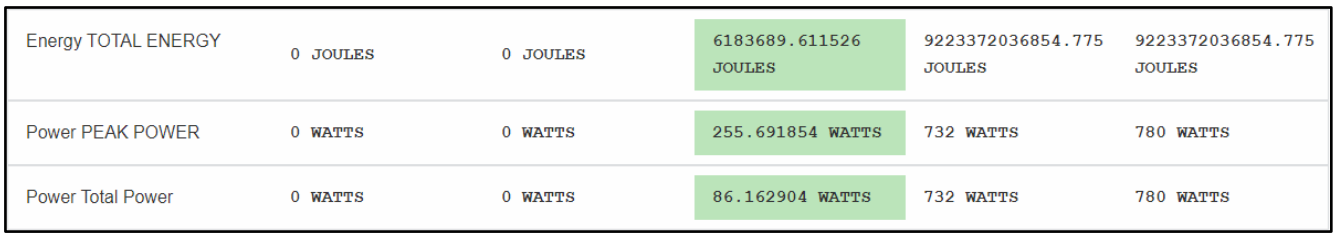


Figure 122: BMC power sensors

4.2.3.1.1 Power sensors Reset

The “Total Energy” and “Peak Power” sensors can be reset whenever necessary, allowing the user to measure the power consumption of the DM1904 chassis during a controlled period of time. Using the reset, the user can control the starting point of the power monitoring and then use the BMC web GUI to read the accumulated power consumed by the whole chassis from the reset moment until the current time.

The reset function for the power sensors is available in the “Firmware” menu of the BMC, as shown in Figure 123.

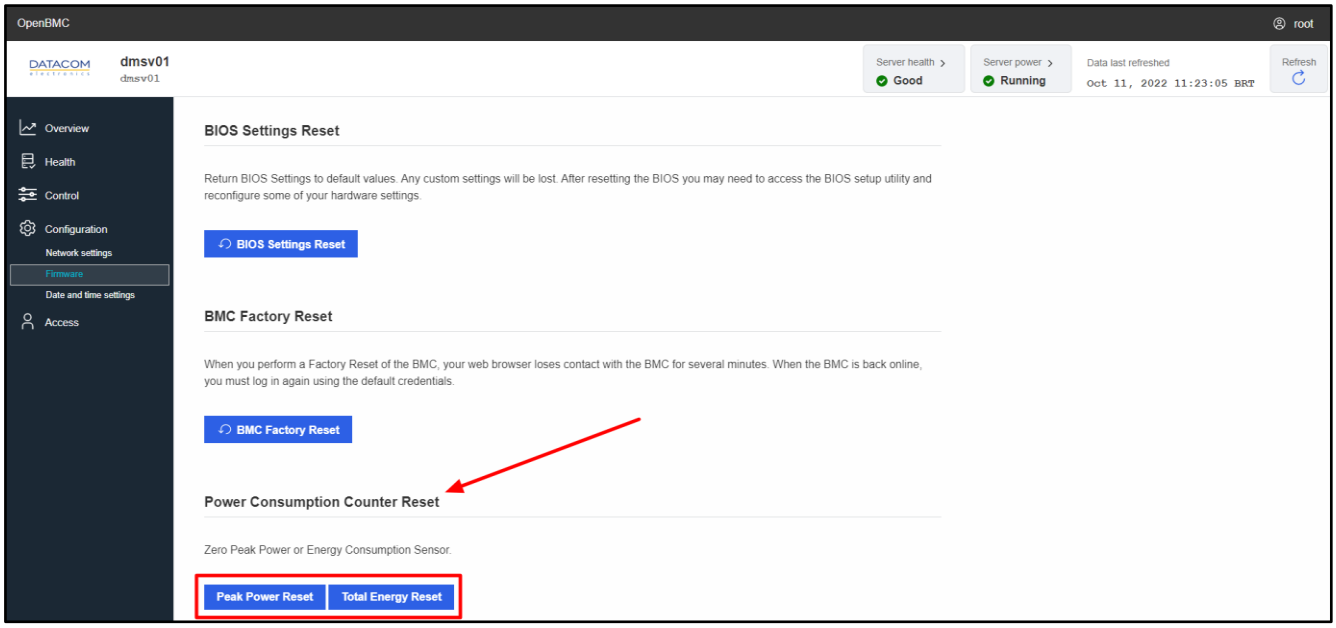


Figure 123: Reset function for the power sensors

4.3 Control Menu

4.3.1 Reboot BMC

The “Reboot BMC” menu of the Supervisory Board BMC is identical to the corresponding menu from the DM-SV01 server BMC. Therefore, refer to section “2.3.3 Reboot BMC” for details regarding this functionality.

4.4 Configuration Menu

4.4.1 Network Settings

This option may be used to configure the network settings of the Supervisory Board BMC.

4.4.1.1 Common Settings

The following Ethernet interface is available for configuration in the Supervisory Board:

- **eth0:** it is the default out-of-band management interface of the Supervisory Board BMC. It can be accessed by means of the dedicated Ethernet port present in the front panel of the DM1904 chassis.

For additional information regarding the Supervisory Board BMC Ethernet port and how to connect to it, please refer to the Supervisory Board section in the “DM-SV01 Product Manual” (1).

The user can select which interface is going to be configured by means of the drop down box shown in the Figure 124. For the Supervisory Board BMC, there is only the interface “eth0” to be selected.

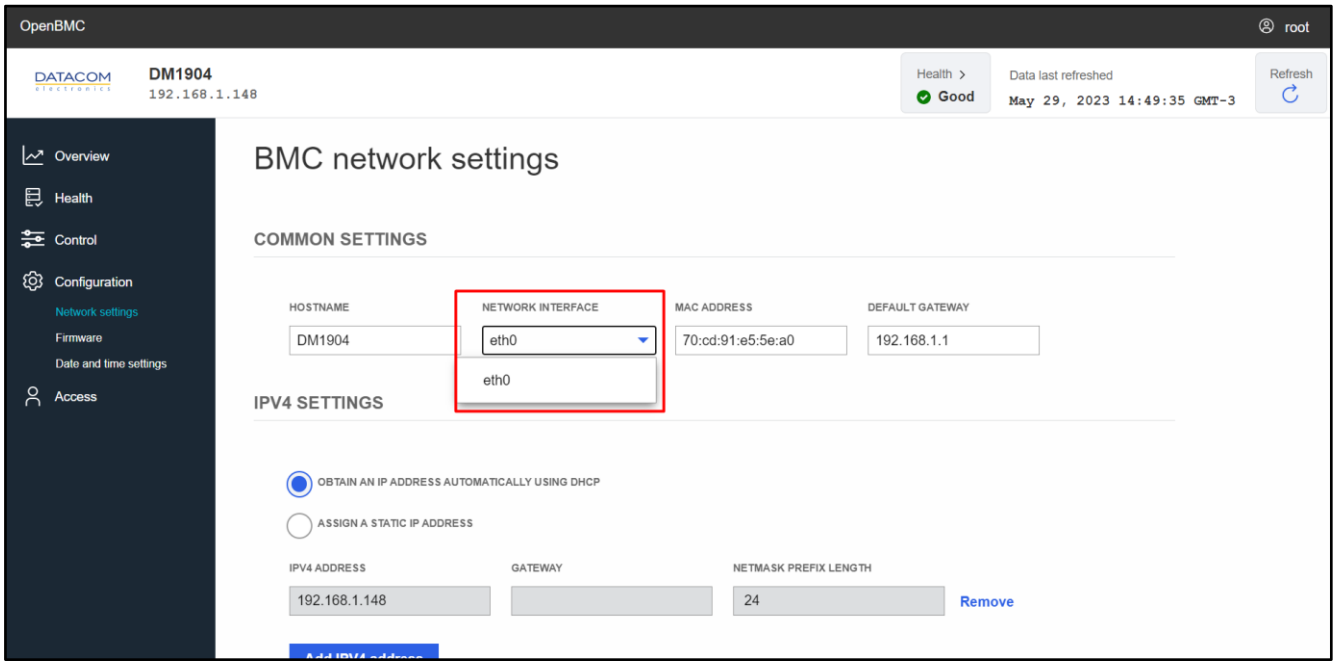


Figure 124: BMC network settings - selecting interface

By default, the eth0 interface is configured as a DHCP client, so it can get the respective IP address automatically.

Besides selecting the network interface to be configured, the following common settings are available:

- **Hostname:** configures the hostname of the BMC, which can be used to access it through the web browser or to connect through SSH.
- **Mac Address:** read only option, it is used to check the MAC address of the selected network interface.
- **Default gateway:** read only option, it is used to check the Default Gateway that is being used by the selected network interface.

4.4.1.2 IPV4 Settings

The settings from this section are specific for the Ethernet port eth0. There are two options available for configuration:

- Configure the network interface as a DHCP client to receive the IP address automatically (default option).
- Assign a static IP address to the network interface.

The selection is done by means of the checkboxes shown below:

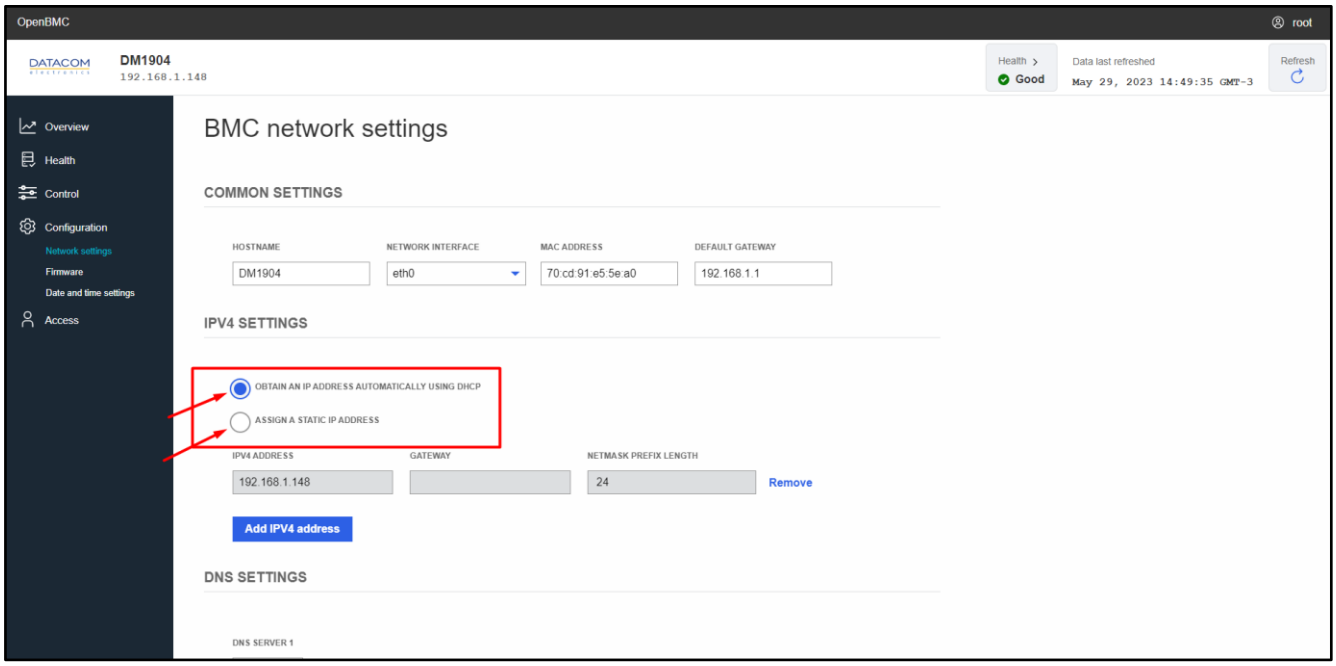


Figure 125: BMC network settings - configuring DHCP or static IPv4 settings

When the option “Assign a static IP address” is selected, the user is prompted to fulfill the following options:

- IPv4 address: in the format “111.111.111.111”.
- Gateway: in the format “111.111.111.111”.
- Netmask prefix length: integer number from “1” up to “32”.

Figure 126: BMC network settings - configuring static IPv4 entries

The operation can be confirmed by clicking on the “Save Settings” button (step 3 in Figure 127).

4.4.1.3 DNS Settings

This option allows the user to manually configure one or more DNS (Domain Name Server) servers. The procedure for adding a DNS address is pretty straightforward. The user just needs to click on the “Add DNS server” button (step 1 in Figure 127) and then fill the IP address inside the desired text box (step 2 in Figure 127). The operation can be repeated for adding more DNS server addresses if needed. The operation can be confirmed by clicking on the “Save Settings” button (step 3 in Figure 127).

Figure 127: BMC network settings - configuring DNS server

4.4.2 Firmware

4.4.2.1 Current FW versions

In the “Firmware” menu, the user can view the current version of the BMC firmware, as well as to perform a FW update.

The image below shows an example of the FW menu section that shows the current FW version of the BMC running on the system.

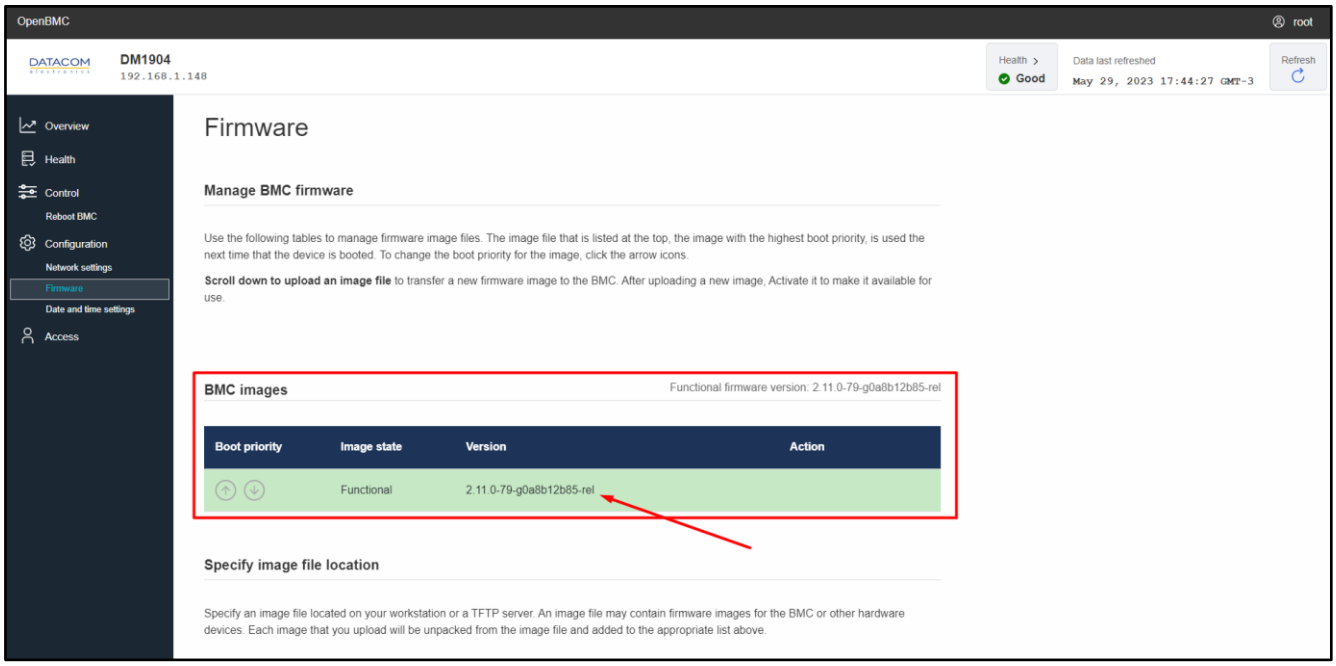


Figure 128: Firmware menu - current FW version

4.4.2.2 BMC FW update process

The FW update process for the Supervisory Board BMC is composed of two main steps: FW image upload and FW activation.

4.4.2.2.1 FW image upload

The section “Specify image file location” is used to upload the BMC image file for performing the FW update. There are two options available:

- **Option 1:** Upload image file from workstation: the file is uploaded by clicking on the “Choose a file” button, selecting the file image from the workstation and then clicking on “Upload firmware”.

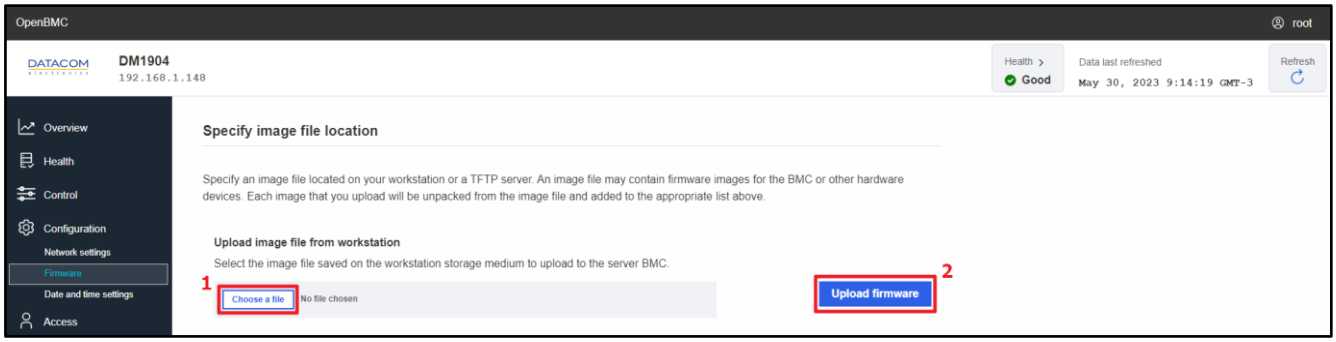


Figure 129: Firmware menu - choosing a file for performing the FW update

- **Option 2:** Download an image file from the TFTP server: the user specifies the TFTP server IP address and the exact file name and then clicks on “Download firmware”.

Figure 130: Firmware menu - configuring TFTP server for file transfer

Once the file is uploaded using any of the means described above, the system checks the image and prepares it for the update.

The BMC checks if the image is valid and signed and pops up a message in the right top corner of the screen indicating if the upload is accepted or not. Once the image is uploaded and properly accepted, the information about the new FW image is placed in the respective image field.

Please consult the Datacom sales team whenever necessary for checking if there is an image available for the update.

The Figure 131 shows an example of a BMC FW image already uploaded and ready for the update in the Supervisory Board.

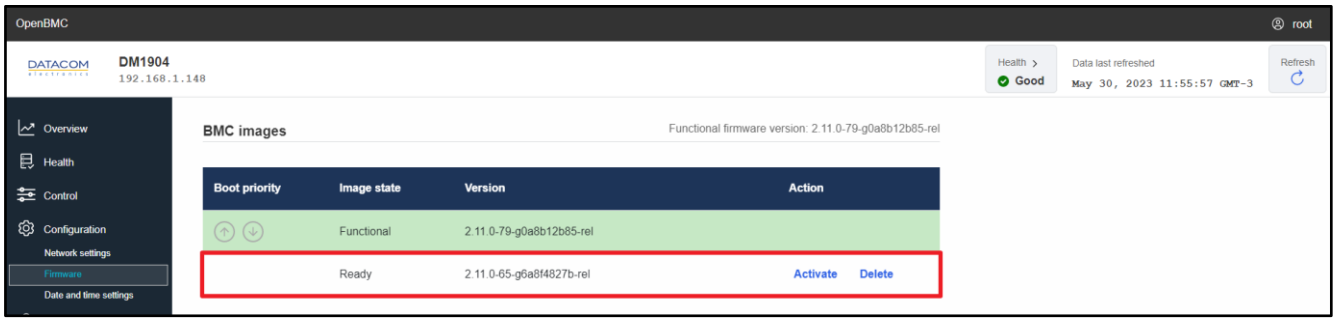


Figure 131: Supervisory Board BMC FW ready for the update

4.4.2.2.2 FW activation

After uploading the FW image by following the steps described in section “4.4.2.2.1 FW image upload”, the user can start the firmware update process by clicking on the “Activate” button.

The Figure 132 shows an example of FW update, where a BMC image for the Supervisory Board has been successfully uploaded and is ready to be activated.

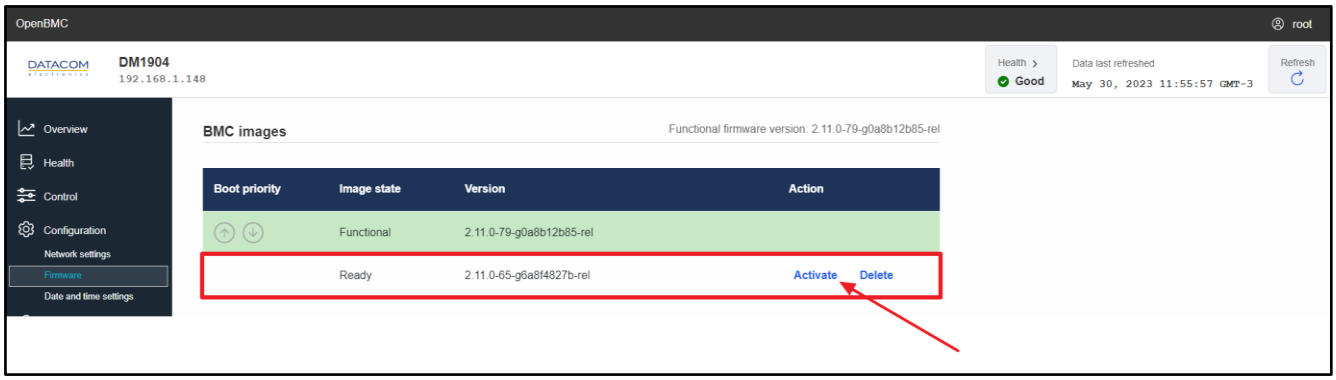


Figure 132: Supervisory Board BMC FW activation

When clicking on the “Activate” button, a confirmation message is displayed on the screen and the user may choose between two options:

- **Activate Firmware File Without Rebooting BMC:** the FW image will be updated, but the BMC will not be rebooted. When this option is selected, any BMC FW change will not take effect until the next BMC reboot is performed.
- **Activate Firmware File and Automatically Reboot BMC (recommended):** the FW image will be updated and the BMC will be rebooted automatically. When this option is selected, any BMC FW change will take effect immediately after the automatic reboot is performed.

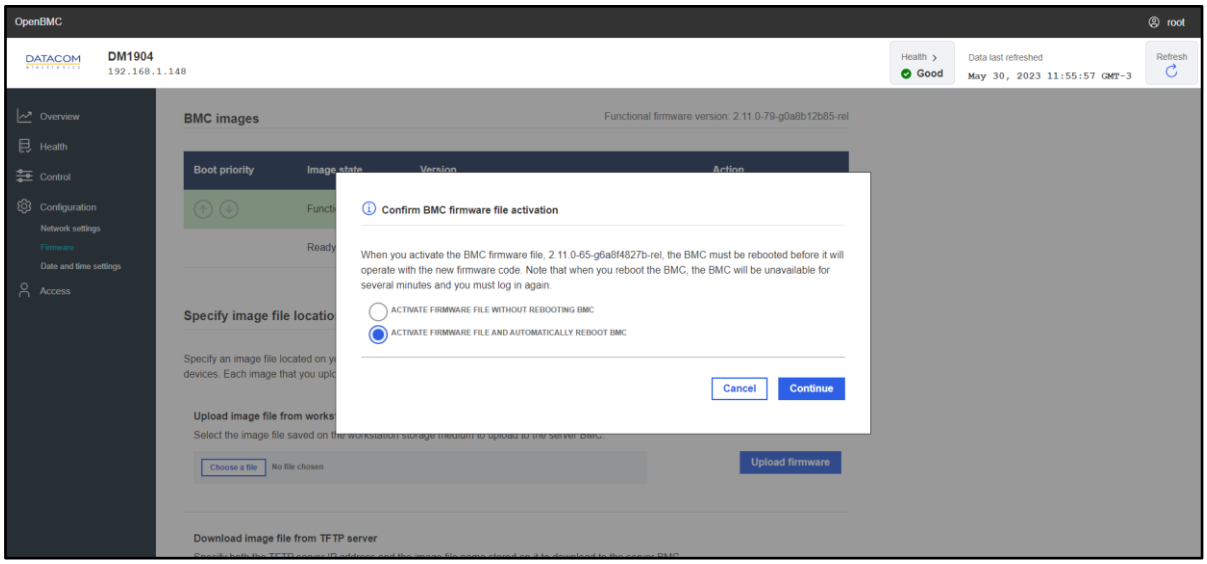


Figure 133: BMC FW activation confirmation

Important: The BMC reboot process does not interfere with the DM1904 power supplies operation. The BMC operation is independent of the PSUs power, so the BMC can be safely rebooted while the PSUs keep feeding the chassis normally.

Important: The user will lose access to the BMC web management GUI while the BMC reboot is being performed, but the access will be restored as soon as the BMC reboot is completed. All the network settings will be preserved during the BMC FW update process, so the user will keep having access to the BMC after the reboot.

4.4.2.3 BMC Factory Reset - Supervisory Board

In the “Firmware” menu, there is also an additional button that is used to reset the BMC settings to factory default. This option can only be set by users with administrator privileges.

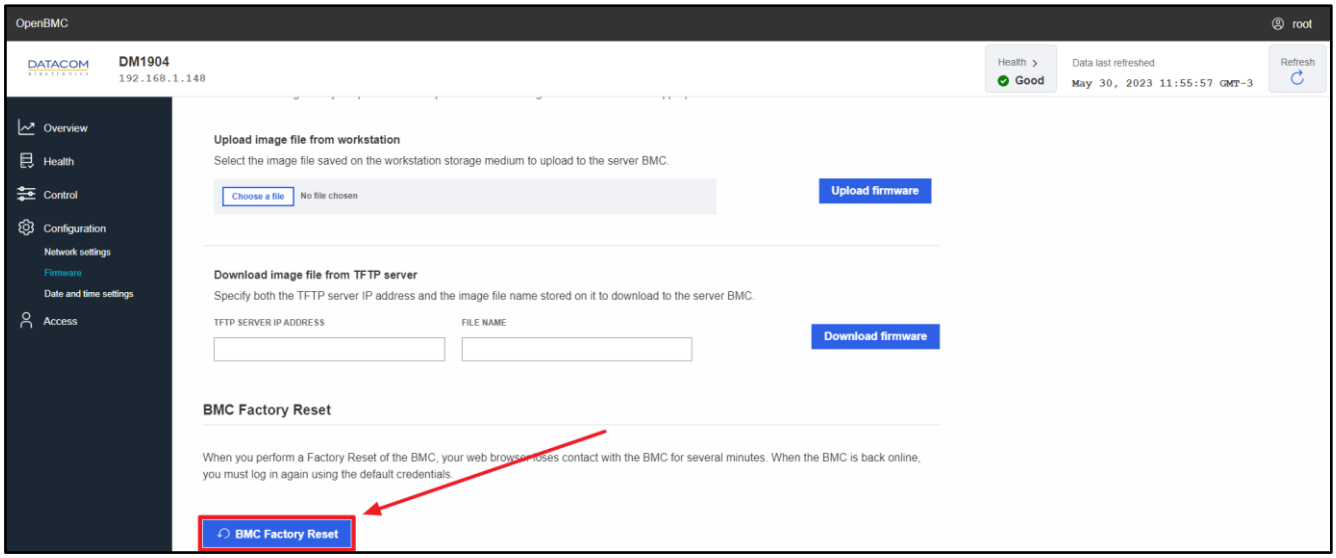


Figure 134: Supervisory Board BMC factory reset

Important: care should be taken before applying the factory reset to the BMC. The user will lose access to the BMC for some minutes when this action is taken, and all the BMC settings will be reset to their default values, including the users and network settings. Therefore, after the factory reset is complete, the user will be able to access the BMC by sending the IP address through a DHCP server and logging in using the default credentials:

- User: root
- Password: 0penBmc (the first digit is the number “zero”)

4.4.3 Date and time settings

The “Date and time settings” menu of the Supervisory Board BMC is identical to the corresponding menu from the DM-SV01 server BMC. Therefore, refer to section “2.4.3 Date and time settings” for details regarding this functionality.

4.5 Access Menu

4.5.1 LDAP

The “LDAP” menu of the Supervisory Board BMC is identical to the corresponding menu from the DM-SV01 server BMC. Therefore, refer to section “2.5.1 LDAP” for details regarding this functionality.

4.5.2 Local Users

The “Local users” menu of the Supervisory Board BMC is identical to the corresponding menu from the DM-SV01 server BMC. Therefore, refer to section “2.5.2 Local users” for details regarding this functionality.

4.5.3 SSL certificates

The “SSL certificates” menu of the Supervisory Board BMC is identical to the corresponding menu from the DM-SV01 server BMC. Therefore, refer to section “2.5.3 SSL certificates” for details regarding this functionality.

5 Supervisory Board Redfish API

The Redfish API is available at the Supervisory Board BMC as a standard way to manage several system resources using the HTTP. The Supervisory Board redfish schema can be accessed by a web browser by means of the URL below:

- https://<BMC_IP>/redfish/v1

After accessing the link above, the user is able to navigate through the links of the Supervisory Board redfish interface in the web browser and access the resources available at the BMC, such as sensors, system inventory data, etc.

5.1 HTTP Methods

The Supervisory Board BMC provides a Redfish RESTful API similar to the interface available at the DM-SV01 server.

Details regarding the API, including its respective methods and responses, can be found at “3.1 HTTP Methods” and “3.2 HTTP responses”.

The section “5.2 Using Redfish with RESTful APIs” lists several management resources and describes how to interact with each one of them.

5.2 Using Redfish with RESTful APIs

The user can send commands to redfish by using a standard RESTful API. The examples shown in this document are using Postman, but the procedure is similar when using other APIs.

The table below lists all the requests available in the BMC from the Supervisory Board. Every request has a specific section explaining its functionality and an example of how to perform the operation.

Function	Method	Section
BMC Login	POST	5.2.1 Session Login
BMC Logout	DELETE	5.2.3 Session Logout
Inventory - Supervisory Board	GET	5.2.4.1 Supervisory Board Inventory
Power sensors	GET	5.2.5.1 Power Sensors
Temperature sensors	GET	5.2.5.2 Temperature Sensors
System Total Power Consumption	GET	5.2.5.3.2 Power Consumption
System Peak Power	GET	5.2.5.3.3 Peak Power
Reset Peak Power sensor	POST	5.2.5.4 Peak Power sensor reset
Reset Energy sensor	POST	5.2.5.5 Energy sensor reset
Network settings	PATCH	5.2.6 Network Settings
Configure Open LDAP	PATCH	5.2.7 LDAP Configuration
Configure LDAP - Active Directory	PATCH	5.2.7 LDAP Configuration
Configure LDAP - Role Groups	PATCH	5.2.7 LDAP Configuration
Change root password	PATCH	5.2.8 Users Management
Add new BMC user	POST	5.2.8 Users Management
Change BMC user role	PATCH	5.2.8 Users Management
Change BMC user password	PATCH	5.2.8 Users Management
Delete BMC user	DELETE	5.2.8 Users Management
Update BMC FW	POST	5.2.9.1 Update BMC Firmware
View log entries	GET	5.2.10 Logging
Clear log entries	POST	5.2.10 Logging
BMC reboot	POST	5.2.11 BMC Reset
BMC factory reset	POST	5.2.11 BMC Reset

5.2.1 Session Login

The “Session Login” procedure of the Supervisory Board BMC is identical to the corresponding Redfish standard from the DM-SV01 server. Therefore, refer to section “3.3.1 Session Login” for details regarding this functionality.

5.2.2 Using the X-Auth-Token

The “Using the X-Auth-Token” procedure of the Supervisory Board BMC is identical to the corresponding Redfish standard from the DM-SV01 server. Therefore, refer to section “3.3.2 Using the X-Auth-Token” for details regarding this functionality.

5.2.3 Session Logout

The “Session Logout” procedure of the Supervisory Board BMC is identical to the corresponding Redfish standard from the DM-SV01 server. Therefore, refer to section “3.3.3 Session Logout” for details regarding this functionality.

5.2.4 System Inventory

5.2.4.1 Supervisory Board Inventory

Using a GET request, it is possible to retrieve inventory information from the DM1904 Supervisory Board.

Function	Inventory - Supervisory Board
Operation	GET
URI	https://<BMC_IP>/redfish/v1/Chassis/supervisoryboard
Payload	none
Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Please, see the example below.

As an example, the excerpt below shows the data provided by the Supervisory Board using the redfish GET request. The user can check relevant information such as model, part number, serial number, etc.


```
{
   "@odata.id": "/redfish/v1/Chassis/supervisoryboard",
   "@odata.type": "#Chassis.v1_14_0.Chassis",
   "Actions": {
```

```
    "#Chassis.Reset": {
        "@Redfish.ActionInfo": "/redfish/v1/Chassis/supervisoryboard/ResetActionInfo",
        "target": "/redfish/v1/Chassis/supervisoryboard/Actions/Chassis.Reset"
    }
},
"ChassisType": "RackMount",
"Id": "supervisoryboard",
"Links": {
    "ComputerSystems": [
        {
            "@odata.id": "/redfish/v1/Systems/system"
        }
    ],
    "ManagedBy": [
        {
            "@odata.id": "/redfish/v1/Managers/bmc"
        }
    ]
},
"Manufacturer": "Datacom",
"Model": "DM1904 - Supervisory Board",
"Name": "supervisoryboard",
"PCIeDevices": {
    "@odata.id": "/redfish/v1/Systems/system/PCIeDevices"
},
"PartNumber": "750.0653.50",
"Power": {
    "@odata.id": "/redfish/v1/Chassis/supervisoryboard/Power"
},
"PowerState": "Off",
"Sensors": {
    "@odata.id": "/redfish/v1/Chassis/supervisoryboard/Sensors"
},
"SerialNumber": "5731772",
"Status": {
    "Health": "OK",
    "HealthRollup": "OK",
    "State": "StandbyOffline"
},
"Thermal": {
    "@odata.id": "/redfish/v1/Chassis/supervisoryboard/Thermal"
}
```

```
}
```

Once the operation is successful, it returns the response “200 OK” and the inventory information is retrieved.

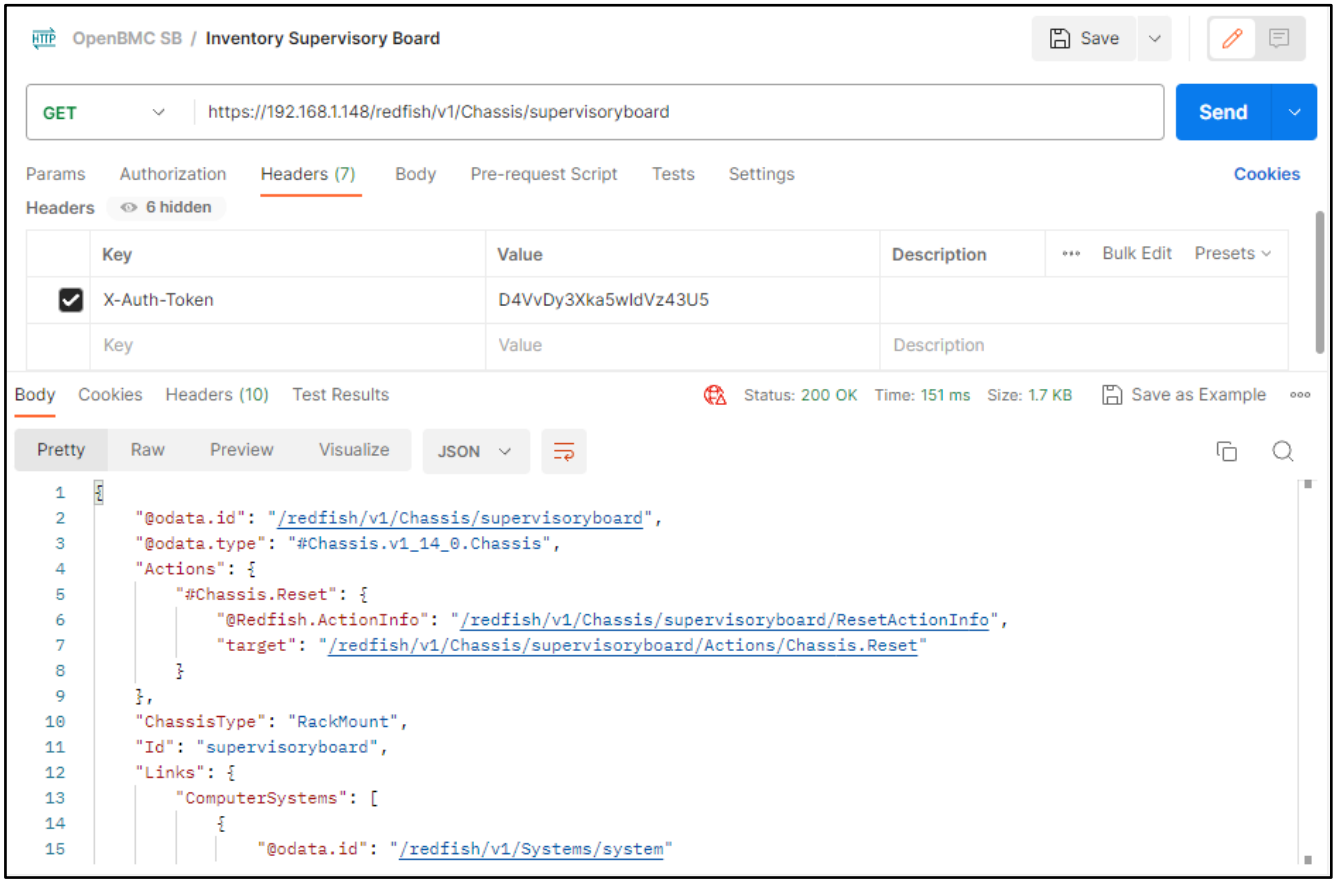


Figure 135: Redfish - Supervisory Board inventory

5.2.5 Sensors

The power and temperature sensors available in the DM1904 chassis can be read by means of the redfish interface. Details regarding the functionality of the DM1904 sensors can be found in section “4.2.3 Sensors”.

5.2.5.1 Power Sensors

Using a GET request, it is possible to retrieve information about the power sensors from the DM1904 chassis.

Function	Power sensors
Operation	GET

URI	https://<BMC_IP>/redfish/v1/Chassis/chassis/Power
Payload	none
Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Please, see the example below.

As an example, the excerpt below shows the data provided by the Supervisory Board using the redfish GET request. The user can check power and voltage measurements from all the power supplies available in the DM1904 chassis, as well as some internal sensors from the Supervisory Board itself.


```
{
   "@odata.id": "/redfish/v1/Chassis/chassis/Power",
   "@odata.type": "#Power.v1_5_2.Power",
   "Id": "Power",
   "Name": "Power",
   "PowerControl": [
      {
         "@odata.id": "/redfish/v1/Chassis/chassis/Power#/PowerControl/0",
         "@odata.type": "#Power.v1_0_0.PowerControl",
         "MemberId": "0",
         "Name": "Chassis Power Control",
         "PowerConsumedWatts": 85.125,
         "PowerLimit": {
            "LimitException": "NoAction",
            "LimitInWatts": null
         },
         "Status": {
            "Health": "OK",
            "State": "Enabled"
         }
      }
   ],
   "Redundancy": [],
   "Voltages": [
      {
         "@odata.id": "/redfish/v1/Chassis/chassis/Power#/Voltages/0",
         "@odata.type": "#Power.v1_0_0.Voltage",
         "MaxReadingRange": null,
         "MemberId": "VIN_PSU_1",
         "MinReadingRange": null,
```

```
        "Name": "VIN PSU 1",
        "ReadingVolts": 119.25,
        "Status": {
            "Health": "OK",
            "State": "Enabled"
        }
    },
    {
        "@odata.id": "/redfish/v1/Chassis/chassis/Power#/Voltages/1",
        "@odata.type": "#Power.v1_0_0.Voltage",
        "MaxReadingRange": null,
        "MemberId": "VIN_PSU_3",
        "MinReadingRange": null,
        "Name": "VIN PSU 3",
        "ReadingVolts": 119.0,
        "Status": {
            "Health": "OK",
            "State": "Enabled"
        }
    },
    {
        "@odata.id": "/redfish/v1/Chassis/chassis/Power#/Voltages/2",
        "@odata.type": "#Power.v1_0_0.Voltage",
        "MaxReadingRange": null,
        "MemberId": "VIN_PSU_4",
        "MinReadingRange": null,
        "Name": "VIN PSU 4",
        "ReadingVolts": 0.0,
        "Status": {
            "Health": "OK",
            "State": "Enabled"
        }
    },
    {
        "@odata.id": "/redfish/v1/Chassis/chassis/Power#/Voltages/3",
        "@odata.type": "#Power.v1_0_0.Voltage",
        "MaxReadingRange": null,
        "MemberId": "VOUT_PSU_1",
        "MinReadingRange": null,
        "Name": "VOUT PSU 1",
        "ReadingVolts": 12.025,
        "Status": {
```

```
            "Health": "OK",
            "State": "Enabled"
         }
      },
      {
         "@odata.id": "/redfish/v1/Chassis/chassis/Power#/Voltages/4",
         "@odata.type": "#Power.v1_0_0.Voltage",
         "MaxReadingRange": null,
         "MemberId": "VOUT_PSU_3",
         "MinReadingRange": null,
         "Name": "VOUT PSU 3",
         "ReadingVolts": 12.255,
         "Status": {
            "Health": "OK",
            "State": "Enabled"
         }
      },
      {
         "@odata.id": "/redfish/v1/Chassis/chassis/Power#/Voltages/5",
         "@odata.type": "#Power.v1_0_0.Voltage",
         "MaxReadingRange": null,
         "MemberId": "VOUT_PSU_4",
         "MinReadingRange": null,
         "Name": "VOUT PSU 4",
         "ReadingVolts": 0.0,
         "Status": {
            "Health": "OK",
            "State": "Enabled"
         }
      }
   ]
}
```


Once the operation is successful, it returns the response “200 OK” and the sensors information is retrieved.

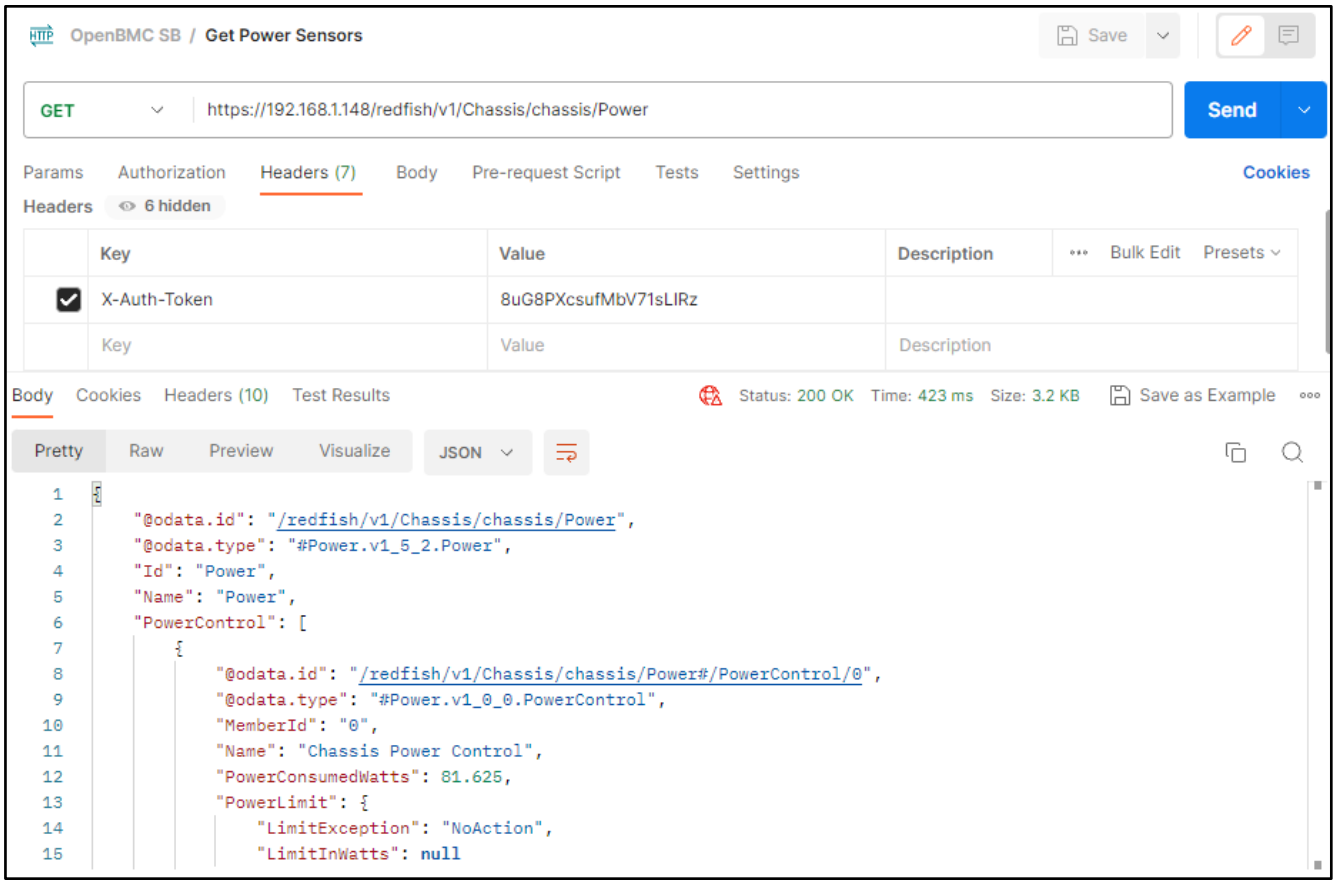


Figure 136: Redfish - Power Sensors

5.2.5.2 Temperature Sensors

Using a GET request, it is possible to retrieve information about the temperature sensors of the power supplies plugged in the DM1904 chassis.

Function	Temperature sensors
Operation	GET
URI	https://<BMC_IP>/redfish/v1/Chassis/chassis/Thermal
Payload	none
Header	X-Auth-Token: “<token>”
Expected response	200 OK
Reply	Please, see the example below.

As an example, the excerpt below shows the data provided by the Supervisory Board when using the redfish GET request. The user can check the measurements from all the temperature sensors from the power supplies available in the chassis.

```
{
   "@odata.id": "/redfish/v1/Chassis/chassis/Thermal",
   "@odata.type": "#Thermal.v1_4_0.Thermal",
   "Fans": [
      {
         "@odata.id": "/redfish/v1/Chassis/chassis/Thermal#/Fans/0",
         "@odata.type": "#Thermal.v1_3_0.Fan",
         "MaxReadingRange": -1,
         "MemberId": "FAN_PSU_1",
         "MinReadingRange": 1,
         "Name": "FAN PSU 1",
         "Reading": 4608,
         "ReadingUnits": "RPM",
         "Status": {
            "Health": "OK",
            "State": "Enabled"
         }
      },
      {
         "@odata.id": "/redfish/v1/Chassis/chassis/Thermal#/Fans/1",
         "@odata.type": "#Thermal.v1_3_0.Fan",
         "MaxReadingRange": -1,
         "MemberId": "FAN_PSU_3",
         "MinReadingRange": 1,
         "Name": "FAN PSU 3",
         "Reading": 3664,
         "ReadingUnits": "RPM",
         "Status": {
            "Health": "OK",
            "State": "Enabled"
         }
      },
      {
         "@odata.id": "/redfish/v1/Chassis/chassis/Thermal#/Fans/2",
         "@odata.type": "#Thermal.v1_3_0.Fan",
         "MaxReadingRange": -1,
         "MemberId": "FAN_PSU_4",
         "MinReadingRange": 1,
         "Name": "FAN PSU 4",
         "Reading": 0,
         "ReadingUnits": "RPM",
         "Status": {
```

```
            "Health": "OK",
            "State": "Enabled"
        }
    }
],
"Id": "Thermal",
"Name": "Thermal",
"Redundancy": [],
"Temperatures": [
    {
        "@odata.id": "/redfish/v1/Chassis/chassis/Thermal#/Temperatures/0",
        "@odata.type": "#Thermal.v1_3_0.Temperature",
        "MaxReadingRangeTemp": null,
        "MemberId": "TEMP_PSU_1",
        "MinReadingRangeTemp": null,
        "Name": "TEMP PSU 1",
        "ReadingCelsius": 26.0,
        "Status": {
            "Health": "OK",
            "State": "Enabled"
        }
    },
    {
        "@odata.id": "/redfish/v1/Chassis/chassis/Thermal#/Temperatures/1",
        "@odata.type": "#Thermal.v1_3_0.Temperature",
        "MaxReadingRangeTemp": null,
        "MemberId": "TEMP_PSU_3",
        "MinReadingRangeTemp": null,
        "Name": "TEMP PSU 3",
        "ReadingCelsius": 31.0,
        "Status": {
            "Health": "OK",
            "State": "Enabled"
        }
    },
    {
        "@odata.id": "/redfish/v1/Chassis/chassis/Thermal#/Temperatures/2",
        "@odata.type": "#Thermal.v1_3_0.Temperature",
        "MaxReadingRangeTemp": null,
        "MemberId": "TEMP_PSU_4",
        "MinReadingRangeTemp": null,
        "Name": "TEMP PSU 4",
```

```
        "ReadingCelsius": 30.0,
        "Status": {
            "Health": "OK",
            "State": "Enabled"
        }
    }
  ]
}
```


Once the operation is successful, it returns the response “200 OK” and the sensors information is retrieved.

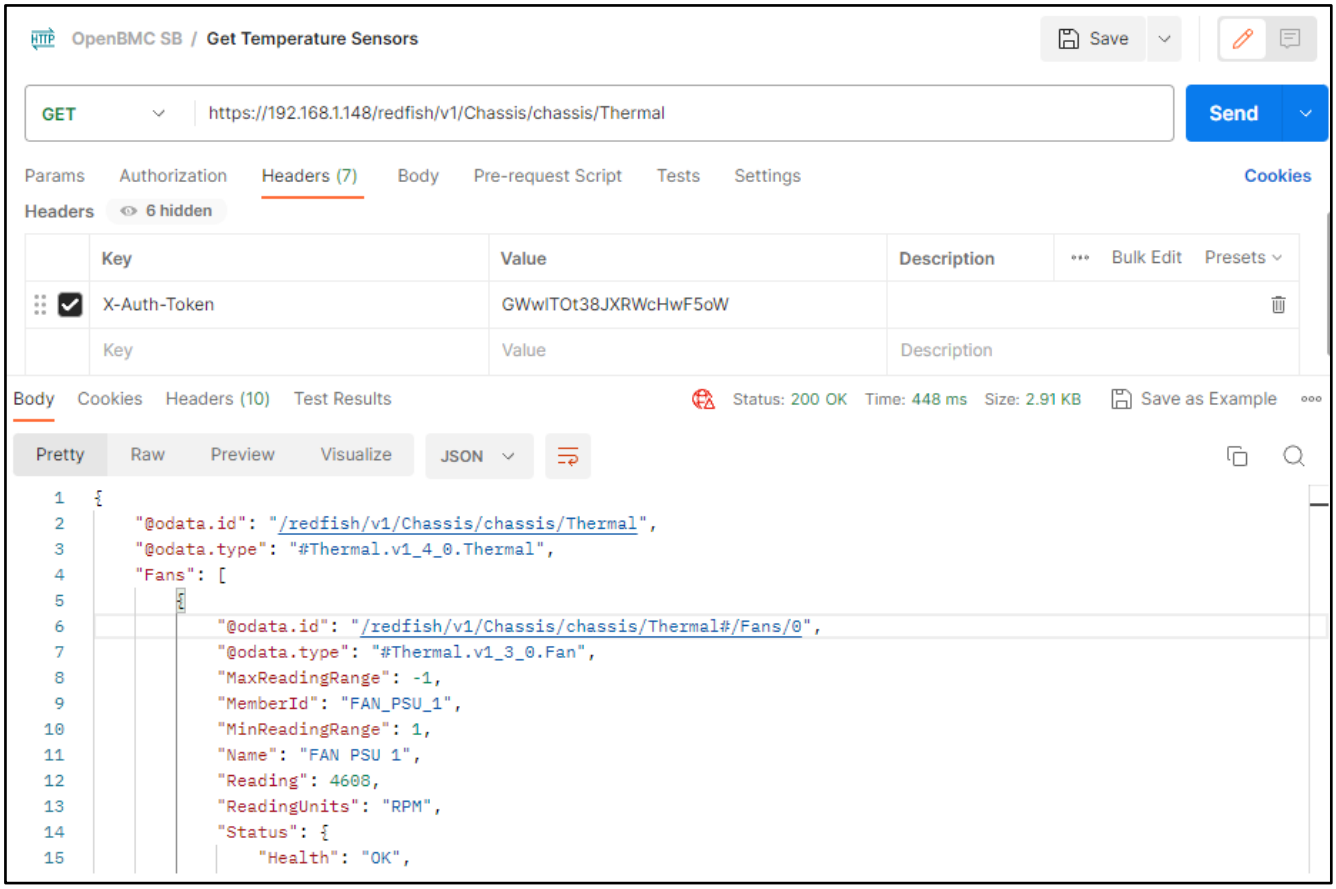


Figure 137: Redfish - Temperature Sensors

5.2.5.3 Power Consumption sensors

5.2.5.3.1 Power Consumption

The “Power consumption” sensor reading procedure of the Supervisory Board BMC is identical to the corresponding Redfish standard from the DM-SV01 server. Therefore, refer to section “3.3.5.3.2 Power Consumption” for details regarding this functionality.

5.2.5.3.2 Peak Power

The “Peak Power” sensor reading procedure of the Supervisory Board BMC is identical to the corresponding Redfish standard from the DM-SV01 server. Therefore, refer to section “3.3.5.3.3 Peak Power” for details regarding this functionality.

5.2.5.4 Peak Power sensor reset

The “Peak Power sensor reset” procedure of the Supervisory Board BMC is identical to the corresponding Redfish standard from the DM-SV01 server. Therefore, refer to section “3.3.5.4 Peak Power sensor reset” for details regarding this functionality.

5.2.5.5 Energy sensor reset

The “Energy sensor reset” procedure of the Supervisory Board BMC is identical to the corresponding Redfish standard from the DM-SV01 server. Therefore, refer to section “3.3.5.5 Energy sensor reset” for details regarding this functionality.

5.2.6 Network Settings

Using a PATCH request, it is possible to configure the network settings of the BMC. The user can configure the **“eth0”** interface, which is the default out-of-band management interface of the BMC. This interface can be accessed by means of the dedicated Ethernet port present in the front panel of the Supervisory Board.

The configuration parameters are inserted in the payload of the request, as shown in the example from the table below. Details regarding the network settings of the BMC can be found in section “4.4.1 Network Settings”.

Function	Network Settings
Operation	PATCH
URI	https://<BMC_IP>/redfish/v1/Managers/bmc/EthernetInterfaces/eth0

Payload	{
	"HostName": "dmsv01",
	"IPv4StaticAddresses": [
	{
	"Address": "192.168.15.101",
	"Gateway": "192.168.15.1",
	"SubnetMask": "255.255.255.0"
	}
	],
	"StaticNameServers": ["8.8.8.8"]
	}

Header	X-Auth-Token: “<token>”
Expected response	204 No Content

Reply	None

Once the operation is successful, it returns the response “204 No Content” and the network settings are applied according to the payload.

Figure 138: Redfish - Configuring network

5.2.7 LDAP Configuration

The “LDAP Configuration” procedure of the Supervisory Board BMC is identical to the corresponding Redfish standard from the DM-SV01 server. Therefore, refer to section “3.3.10 LDAP Configuration” for details regarding this functionality.

5.2.8 Users Management

The “Users Management” procedure of the Supervisory Board BMC is identical to the corresponding Redfish standard from the DM-SV01 server. Therefore, refer to section “3.3.11 Users Management” for details regarding this functionality.

5.2.9 FW Update

The user can update the BMC FW by means of the redfish. Details regarding the FW update can be found in section “2.4.2.2 FW update process - BMC or BIOS”.

5.2.9.1 Update BMC Firmware

The “Update BMC Firmware” procedure of the Supervisory Board BMC is identical to the corresponding Redfish standard from the DM-SV01 server. Therefore, refer to section “3.3.12.1 Update BMC Firmware” for details regarding this functionality.

5.2.10 Logging

The “Logging” procedure of the Supervisory Board BMC is identical to the corresponding Redfish standard from the DM-SV01 server. Therefore, refer to section “3.3.13 Logging” for details regarding this functionality.

5.2.11 BMC Reset

The “BMC Reset” procedure of the Supervisory Board BMC is identical to the corresponding Redfish standard from the DM-SV01 server. Therefore, refer to section “3.3.14 BMC Reset” for details regarding this functionality.

6 References

(1) “DM-SV01 - Product Manual”.

(2) “DM-SV01 - BIOS User Manual”.

(3) RFC 9110 - HTTP Semantics, available at “https://www.rfc-editor.org/rfc/rfc9110.html”.

(4) **John Leung, Intel.** “Introduction and Overview of Redfish”. SNIA, 2017. Available at “https://www.snia.org/educational-library/introduction-and-overview-redfish-2017”.

(5) “Redfish”. DMTF. Available at “https://www.dmtf.org/standards/redfish”.

7 Annex A - Frequently Asked Questions

Q: Why is the Firmware update rejected by the BMC?

A: Updates are digitally signed by Datacom to ensure integrity. Only correctly signed update packages will be accepted. Furthermore, BMC only supports update and downdate, not redate.

Q: Is the VGA output limited to only 800x600?

A: In case no external monitor is found, resolution defaults to 800x600. In order to obtain higher resolutions, please attach an external display. When using Windows OS, it is possible to get higher resolutions by installing the Aspeed driver.

Q: How to select the correct keymap for KVM?

A: Scancodes from the keyboard are sent directly to the host operating system, therefore select the desired keymap within the host operating system.

Q: How to force booting the recovery BMC software?

A: Press and hold the ID Button while applying power to the board. Continue pressing the button until the ID LED blinks for 2 times.

Q: How to update the BMC date and time settings after the initial powerup?

A: Either boot the host system for the BIOS to supply correct time information to the BMC or configure a network time server for the BMC in the settings. A network time server configuration is highly recommended.

Q: Why are there some grayed out sensors in the web interface?

A: These sensors are only able to provide valid data when either the appropriate HW is installed and/or the host system is running.

Q: When changing the IP address of the BMC's management port the old address remains and the new one is added as an extra address. Is this the expected behavior?

A: In order to guarantee continued connectivity the old address remains unchanged until manually removed.

Q: The BMC management port is misconfigured so that it cannot be accessed anymore over the network. How can I recover from this situation?

A: Use the BMC Network Configuration option in the BIOS to configure the network management ports.

This document comprises 163 pages.

Revision History:

Date	Description
2022/09/23	1.0 - First Official Release
2022/10/21	Release 2.0 - Added redfish section and minor updates: - Added Introduction to Redfish (section 1.4) - Added information about Peak Power and Total Energy sensors (section 2.2.3.1.1) - Added a complete explanation of the Redfish API (section 3) - Added FAQ section (Annex A)
2022/11/07	Release 2.1 - Added section to clarify some requirements for configuring an LDAP server. - Added section 2.5.1.3 Instructions for implementing the LDAP server.
2023/06/15	Release 3.0 - Added section describing the process for recovering the BMC password and added general information about the BMC for the DM1904 Supervisory Board. - Added section “1.3 BMC password recovery”. - Added section “4 Supervisory Board BMC”. - Added section “5 Supervisory Board Redfish API”.