Dell Command | Monitor Version 10.6

User's Guide

October 2021
Rev. A01

Notes, cautions, and warnings

ⓘ **NOTE:** A NOTE indicates important information that helps you make better use of your product.

⚠ **CAUTION: A CAUTION indicates either potential damage to hardware or loss of data and tells you how to avoid the problem.**

⚠ **WARNING: A WARNING indicates a potential for property damage, personal injury, or death.**

Contents

Introduction to Dell Command | Monitor 10.6

The Dell Command | Monitor software application enables IT administrators to easily manage fleet inventory, monitor system health, modify BIOS settings, and remotely collect information for deployed Dell client systems.

Active system health state monitoring can help reduce the total cost of system ownership and is part of a holistic approach to managing all networked devices.

Dell Command | Monitor is designed for Dell Enterprise client systems, Dell IoT Gateway systems, and for Dell Embedded PCs.

This document provides an overview of Dell Command | Monitor and its features. For more information about supported Dell systems see Release notes available on **dell.com/support**.

Topics:

What's new in this release for Dell Command | Monitor 10.6

- Support for the following new BIOS attributes:
  - `Intel@ GNA Accelerator`
  - `Multiple Atom Cores`
  - `USB4 CM Mode`
  - `Onboard Unmanaged NIC`
  - `Enable Pre-Boot DMA Support`
  - `Enable OS Kernel DMA Support`
  - `PCIe Resizable Base Address Register (BAR)`
  - `OS Agent Requests`
  - `Enable Microsoft UEFI CA`
  - `Legacy Manageability Interface Access`
  - `Power-on-Self-Test (POST) Automatic Recovery`
- Support for the **ResetBIOSDefaults** method.

Dell Command | Monitor overview

NOTE: Simple Network Management Protocol (SNMP) is not supported for Dell Command | Monitor for Linux.

Dell Command | Monitor manages client systems using the management protocols Common Information Model (CIM) standard and Simple Network Management Protocol (SNMP). This helps to reduce the total cost of system ownership, improves security, and provides a holistic approach to manage all the devices within a network device.

Using CIM you can access Dell Command | Monitor through Web Services for Management Standards (WSMAN).

Dell Command | Monitor contains the underlying driver set that collects client system information from different sources including the BIOS, CMOS, System Management BIOS (SMBIOS), System Management Interface (SMI), operating system, and Application Programming Interface (APIs). Dell Command | Monitor for Windows also collects client system information from Dynamic-Link Library (DLLs), and registry settings. Dell Command | Monitor for Windows retrieves this information through the CIM Object Manager (CIMOM) interface, Windows Management Instrumentation (WMI) stack, or SNMP agent, whereas Dell Command | monitor for Linux retrieves this information through Open Management Infrastructure (OMI) interface.

Dell Command | Monitor enables IT administrators to remotely collect asset information, modify BIOS settings, receive proactive notifications about potential fault conditions, and get alerts for potential security breaches. In the systems running Windows, these alerts are available as events in the NT event log, WMI event, or SNMP traps v1. For the systems running Linux, these alerts are received as Syslog, OMI event, or Application log.

Dell Command | Monitor for Windows can be integrated into a console such as Microsoft System Center Configuration Manager by directly accessing the CIM information, or through other console vendors who have implemented the Dell Command | Monitor integration. Also, you can create custom scripts to target key areas of interest. Sample scripts are available at Dell Knowledge Library Dell Command | Monitor page. You can use these scripts to monitor inventory, BIOS settings, and system health.

NOTE: Default installation does not enable SNMP support. For more information about enabling SNMP support for Dell Command | Monitor for Windows, see Dell Command | Monitor Installation Guide at **dell.com/support**.

Windows SMM Security Mitigations Table (WSMT) Compliance

The Windows (SMM) Security Mitigations Table contains information about the ACPI table that was created for the Windows operating system, which supports Windows virtualization-based security (VBS) features. Dell Command | Monitor is WSMT compatible. This is used for configuring the platform features on Dell Client Systems with WSMT enabled BIOS.

Following is the behavioral change due to WSMT compliance:

Dell Command | Monitor functionalities are available on Dell client platforms which have the compatible version of BIOS supporting WMI/ACPI.

ⓘ **NOTE:** For more information about supported platforms, see Supported Platforms.

Standards and protocols for Dell Command | Monitor 10.6

Dell Command | Monitor is based on CIM standards. The CIM specification details mapping techniques for improved compatibility with management protocols.

Management protocols such as WMI, SNMP, and WSMAN are used for remote monitoring.

ⓘ **NOTE:** Dell Command | Monitor for Windows uses Simple Network Management Protocol (SNMP) to describe several variables of the system.

The Desktop Management Task Force (DMTF) is the industry-recognized standards body that leads the development, adoption, and unification of management standards (including CIM and ASF), and initiatives for desktop, enterprise, and Internet environments.

Use case scenarios using Dell Command | Monitor 10.6

This section describes various use case scenarios of Dell Command | Monitor.

You can use Dell Command | Monitor for:

- Asset management
- Configuration management
- Health monitoring
- Profiles

Topics:

Scenario 1: Asset management

A company that has many Dell systems was not able to maintain accurate inventory information because of changes in the business and IT staff. The Chief Information Officer (CIO), requests a plan for identifying the systems that can be upgraded to the latest version of Windows. This requires an assessment of deployed systems to determine the size, scope, and financial impact of such a project. The information collection involves a significant effort. Deploying IT staff to each client system is expensive in terms end-user interruptions.

Using Dell Command | Monitor on each Dell system, the IT manager can quickly collect information remotely. Using tools such as Microsoft System Center Configuration Manager (SCCM), the IT manager queries each client system over the network and collects information such as CPU type and speed, memory size, hard-drive capacity, BIOS version, and current operating system version. Once the information is collected, it can be analyzed to identify the systems that can be upgraded to the latest version of Windows.

You can also get asset inventory through WSMAN/WinRM command line or any using any CIM client command line.

SCCM integration

You can integrate SCCM with Dell Command | Monitor for Windows by:

- Using the MOF file within Dell Command | Monitor install package, which contains all the Dell Command | Monitor classes and importing to ConfigMgr

  The MOF is located at:

  ```
  C:\Program Files\Dell\Command_Monitor\ssa\omacim\OMCI_SMS_DEF.mof
  ```

- Extending asset reporting capabilities using collections

Scenario 2: Configuration management

A company plans to standardize the client platform and manage each system through its lifecycle. As part of this effort, the company acquires a suite of tools and plans to automate the deployment of a new client operating system using the Preboot Execution Environment (PXE).

The challenge is to modify the BIOS password in each client computer without manually visiting each desktop. With Dell Command | Monitor installed on each client system, the IT department of the company has several options to remotely modify the boot order. The OpenManage Essentials (OME) is a management console that can be integrated with Dell command | Monitor and used to monitor BIOS settings remotely on all client systems. Another option is to write a script (CIM, WinRM/WSMAN/PowerShell/WMIC) that changes the BIOS setting. The script can be remotely delivered over the network and run on each client system.

For more information on Dell Command | Monitor, see Dell Command | Monitor Reference Guide at **dell.com/support**.

Standardized configurations can provide significant cost savings for companies of all sizes. Many organizations deploy standardized client systems, but few manage the system configuration throughout the life of the computer. With Dell Command | Monitor installed on each client system, the IT department can lock down Legacy ports to prevent the use of unauthorized peripherals, or enable Wake On LAN (WOL) to revive the system from a sleep state during non-peak hours to perform systems management tasks.

Scenario 3: Health monitoring

A user receives read error messages while trying to access certain files on the client-system hard drive. The user reboots the system and the files now appear to be accessible. The user disregards the initial problem because it appears to have resolved itself. Meanwhile, Dell Command | Monitor queries the hard drive with the problem for a predicted failure and passes a Self-Monitoring, Analysis and Reporting Technology (SMART) alert to the management console. It also displays the SMART error to the local user. The alert indicated that several read/write errors are occurring in the hard drive. The IT department of the company recommended that the user must make a backup of critical data files immediately. A service technician is dispatched with a replacement drive.

The hard drive is replaced before it fails, preventing user downtime, a help desk call, and a technician trip to the desktop to diagnose the problem.

Monitoring system alerts through operating system Event Viewer, Syslog, or CIM indication

Dell Command | Monitor supports monitoring events through the following procedures:

- Pulling the log through CIM class DCIM_LogEntry.
- Monitoring CIM indication through DCIM_AlertIndication class.
- (only for Dell Command | Monitor for Windows) Monitoring events through Simple Network Management Protocol (SNMP) and Windows event viewer.
- (only for Dell Command | Monitor for Linux) Monitoring through Syslog.

For more information on Dell Command | Monitor, see Dell Command | Monitor Reference Guide at **dell.com/support**.

Scenario 4: Profiles

ⓘ **NOTE:** DMTF profiles are implemented for Dell Command | Monitor for Windows only.

IT administrators are required to manage client systems in multi-vendor and distributed enterprise environments. They face challenges as they must master a diverse set of tools and applications while managing several desktop and mobile client systems in various networks. To reduce the cost of these requirements and represent the provided management data, the industry-standard Distributed Management Task Force (DMTF) and Data Center Infrastructure Management (DCIM-OEM) profiles are implemented in Dell Command | Monitor. Some of the DMTF profiles are explained in this guide.

For more information on Dell Command | Monitor, see Dell Command | Monitor Reference Guide at **dell.com/support**.

Asset profile

Warranty Status on endpoint device:

- Determine the status of the warranty by enumerating or getting the instance of the class **DCIM_AssetWarrantyInformation**.
- Check if the warranty status can be determined using the properties **WarrantyStartDate** and **WarrantyEndDate** of the class **DCIM_AssetWarrantyInformation**.

> **NOTE:** Prerequisite to DCIM_AssetWarrantyInformation is that you must have a working Internet connection. If you are running Dell Command | Monitor behind a proxy server, ensure that the proxy settings are configured correctly.

To get more information about the warranty status of the peripherals:

1. Go to Dell.com/support
2. Verify your country or region in the Choose a Country/Region drop-down list at the bottom of the page
3. Select support category - Warranty and Contracts
4. Provide the appropriate service tag of your system

- Disable warranty feature and subsequent refresh calls.
- Pull warranty information on-demand.

> **NOTE:** Warranty information is automatically updated every 15 days. In case of recent warranty status, the warranty information enumerated may not be same as the one on the Dell support site.

Battery profile

- Determine the status of the battery by enumerating or getting the instance of the class **DCIM_Battery**.
- Determine the estimate run time and see the estimated remaining charge.
- Check if the health information of the battery can be determined using the properties Operational Status andHealthState of the class **DCIM_Battery**.
- Get additional information about the health of a battery using **DCIM_Sensor.CurrentState** property or the **CIM_NumericSensor.CurrentState** property.
- Determine the battery location and the battery ePPID by using the **IdentifyingDescriptions** and **OtherIdentifyingInfo** properties of the class **DCIM_Battery**.

DCIM_Battery

To get the information about the battery ePPID value for a battery element. Open a PowerShell prompt with the administrative privileges and run the following command `Get-CimInstance -Namespace root/dcim/sysman -Classname DCIM_Battery |Select ElementName, OtherIdentifyingInfo, IdentifyingDescriptions`.

> **NOTE:** Battery ePPID value is not dynamic and if the battery is replaced, you must restart the system to reflect the changes in the **DCIM_Battery** instance.

BIOS management profile

- Determine the BIOS version by enumerating the instance of the class **DCIM_BIOSElement**.
- Check whether BIOS attribute values can be modified or not. Get the instance of the class, **DCIM_BIOSEnumeration**. The attribute can be modified if the property **IsReadOnly** is set to FALSE.
- Set the system password (SystemPwd). Run the **DCIM_BIOSService.SetBIOSAttributes()** method and set the SystemPwd to AttributeName and password value to AttributeValue parameters.
- Set the BIOS or Admin password (AdminPwd). Run the **DCIM_BIOSService.SetBIOSAttributes()** method and set the AdminPwd to AttributeName and password value to AttributeValue parameters.
- Run the **DCIM_BIOSService.SetBIOSAttributes()** method and specify the AttributeName and AttributeValue parameters.
- To modify a BIOS Attribute when BIOS or Admin password is set, run the **DCIM_BIOSService.SetBIOSAttributes()** method and specify the AttributeName, AttributeValue, and current BIOS password as the AuthorizationToken input parameter.

Boot control

- Change the sequence of the boot items in the Legacy and UEFI boot list.
- Enable or disable the boot items in the Legacy and UEFI boot list.
- Find the current boot configuration by enumerating the instances of the class **DCIM_ElementSettingData** whose **IsCurrent** property is set to **1**. The **DCIM_BootConfigSetting** represents the current boot configuration.

Base desktop mobile

- Determine the system model, service tag, and serial number by enumerating the instance of the class, **DCIM_ComputerSystem**.
- You can use the **DCIM_ComputerSystem.RequestStateChange()** method to set the RequestedState parameter value to **3**. The parameter value 3, turns off the system.
- You can use the **DCIM_ComputerSystem.RequestStateChange()** method to set the **RequestedState** parameter value to **11**. The parameter value 11, reboots the system.
- Determine the power state of the system.
- Determine the number of processors in the system by querying **DCIM_Processor**, instances which are associated with the Central Instance through the **DCIM_SystemDevice** association.
- Get the system time. Run the **DCIM_TimeService.ManageTime()** method and set the **GetRequest** parameter to **True**.
- Check the health status of the managed element.

Log record

- Identify the log name by selecting the **DCIM_RecordLog** instance in which the **ElementName** property corresponds to the log name.
- Find the individual log entries. Get all the instances of **DCIM_LogEntry** that are associated with the given instance of **DCIM_RecordLog** through the **DCIM_LogManagesRecord** association. Sort the instances based on the **RecordID**.
- Check whether record logs are enabled or not by enumerating the instance of the class **DCIM_RecordLog** whose property **Enabledstate** is set to **2** (represents enabled) and **EnabledState** is set to **3** (represents disabled).
- Sort the log records based on the time stamp of the log entry. Get all the instances of **DCIM_LogEntry** that are associated with the given instance of **DCIM_RecordLog** through the **DCIM_LogManagesRecord** association. Sort the instances of **DCIM_LogEntry** based on the **CreationTimeStamp** property value in Last In First Out (LIFO) order.
- Clear logs by running the **ClearLog()** method for the given instance of the **DCIM_RecordLog**.

Physical asset

- Obtain the physical inventory for all the devices in a system.
- Obtain the physical inventory for a system chassis.
- Determine the part number of a failing component.
- Determine whether the slot is empty or not.

System memory profile

- Obtain the memory information of the system.
- Obtain the physical memory information of the system.
- Check the system memory size.
- Check the available system memory size.
- Check the physical system memory size.
- Check the health status of system memory.

5

Using Dell Command | Monitor 10.6

You can view the information that is provided by Dell Command | Monitor by accessing: `root\dcim\sysman (standard)`

Dell Command | Monitor provides the information through classes in these namespaces.

For more information about the classes, see Dell Command | Monitor Reference Guide at **dell.com/support**.

Topics:

Polling interval setting

You can change the polling interval of fan probe, temperature probe, voltage probe, current probe, disk capacity increase/decrease, memory size increase/decrease and number of processors increase/decrease using Dell Command | Monitor.

- For Windows, `dcsbdy32.ini` or `dcsbdy64.ini` file is present at `<Dell Command | Monitor installed location>\omsa\ini`.
- For Linux, `AlertPollingSettings.ini` file is present at `/opt/dell/dcm/conf`.

ⓘ **NOTE:** The numbers in the INI file is multiplied by **23**. The default polling interval for disk capacity and Self-Monitoring, Analysis, and Reporting Technology (SMART) alert is **626** seconds (the real time = 626 X 23 seconds which is approximately 3 hours).

RAID status reporting

Dell Command | Monitor enables the RAID configuration information and monitors the RAID functionality for client systems with hardware and driver support. You can use RAID classes to receive the details about RAID levels, driver information, controller configuration, and controller status. After the RAID configuration is enabled, you can receive alerts for degradation or failure of drives and controllers.

ⓘ **NOTE:** RAID status reporting is supported only for the RAID controllers which work on Common Storage Management Interface (CSMI) version 0.81 compliant drivers. OMCI 8.1 and later versions support monitoring only on the Intel on-chip RAID controller; and from OMCI 8.2 and later versions support Alerting for Intel on-chip RAID controller.

Monitoring the Dell client systems

- Dell Command | Monitor for Windows supports Simple Network Management Protocol (SNMP) for monitoring and managing client systems such as notebooks, desktops, and workstations. The Management Information Base (MIB) file is shared between Dell Command | Monitor and Server Administrator. Dell Command | Monitorfor Windows from version 9.0 has been modified to use an OID that is specific to client OID (10909) for consoles to identify client systems.

For more information about SNMP, see Dell Command | Monitor SNMP Reference Guide at **dell.com/support**.

- Dell Command | Monitor for Linux supports monitoring using WinRM and WSMan commands.

Application log for Dell Command | Monitor for Linux

Dell Command | Monitor for Linux segregates the application logs and alerts for reporting and debugging purpose. The history of the generated alerts and logs for the Dell Command | Monitor application can be viewed in the **dcm_application.log** file available at `/opt/dell/dcm/var/log`.

Configuration file

You can update the configuration file **log.property** available at `/opt/dell/dcm/conf` to apply the desired settings and DEBUG:

NOTE: Restart the OMI server after making any change in the configuration file to apply the changes.

- Log_Level — There are three log levels to segregate the system messages: ERROR, INFO, DEBUG

  The user can change the log level from the configuration file. If the log level is set to DEBUG, the Dell Command | Monitor application log will send all the information in to the specified log file.

  NOTE: The default log level is set to INFO.
- File_Size — The user can specify the maximum size of the **dcm_application.log** file. The default file size is 500 MB.

  NOTE: The File_Size value must be expressed in bytes.
- BackupIndex — The user can specify the rollover count of the **dcm_application.log** file. If the default rollover count is 2, the third backup file will override the oldest file.

Detecting advance format drives

Client systems are transitioning to Advanced Format (AF) drives for larger storage capacity and to address the limitations of 512-byte sector hard drives (HDDs). The hard drives transitioning to 4KB sectors maintain backward compatibility, while the current AF hard drive, known as 512e hard drive, match 512-byte SATA and operate at 4KB. During the transition, you may encounter performance issues such as partition mis-aligned drives in the client systems resulting in failure of sector-based encryption software packages that handle 512e drives. Dell Command | Monitor allows you to determine if the hard drive on a system is 4KB AF drive, which helps to prevent these issues.

Boot configurations

NOTE: Dell Command | Monitor for Linux does not offer the boot configuration capabilities. So this section is not applicable for Dell Command | Monitor for Linux.

A client system can have one of two types of boot configuration:

- Legacy (BIOS)
- UEFI

In Dell Command | Monitor, the boot configuration (Legacy or UEFI) is modeled using the following classes:

- DCIM_ElementSettingData
- DCIM_BootConfigSetting
- DCIM_OrderedComponent
- DCIM_BootSourceSetting
- DCIM_SmartAttributeInfo

NOTE: The terms Boot Configuration and Boot List Type are used interchangeably and convey the same meaning representing Legacy or UEFI.

DCIM_AssetWarrantyInformation

- To query the warranty status on endpoint device, run the following command in the PowerShell prompt:

  ```
  Get-CimInstance -Namespace root/DCIM/SYSMAN -ClassName DCIM_AssetWarrantyInformation
  ```

- To list the warranty entitlements in chronological order of `WarrantyEndDate`, run the following command in the PowerShell prompt:

  ```
  Get-CimInstance -Namespace root/dcim/sysman -ClassName DCIM_AssetWarrantyInformation
  | Sort-Object -Property WarrantyEndDate | Select Name, WarrantyEndDate,
  WarrantyStartDate
  ```

- To disable the warranty feature and subsequent refresh calls, run the following command in the PowerShell prompt:

  ```
  Get-CimInstance -Namespace root/DCIM/SYSMAN -ClassName DCIM_AssetWarrantyInformation|
  Where-Object{$_.InstanceID -eq "Root/MainSystemChassis/COOObject/COOWarranty:0"} |
  Invoke-CimMethod -MethodName DisableWarranty
  ```

- To pull warranty information on-demand, run the following command in the PowerShell prompt:

  ```
  Get-CimInstance -Namespace root/DCIM/SYSMAN -ClassName DCIM_AssetWarrantyInformation|
  Where-Object{$_.InstanceID -eq "Root/MainSystemChassis/COOObject/COOWarranty:0"} |
  Invoke-CimMethod -MethodName RefreshWarranty
  ```

NOTE: Set up for Proxy configuration -

- Default proxy – Dell Command | Monitor selects the default system proxy (set in IE)
- Custom proxy

  DCIM_ApplicationProxySetting class is used to modify the proxy settings for Dell Command | Monitor as per the proxy environment.

DCIM_BootConfigSetting

An instance of **DCIM_BootConfigSetting** represents a boot configuration that is used during the boot process. For example, on client systems, there are two types of boot configurations—Legacy and UEFI. So, **DCIM_BootConfigSetting** has a maximum of two instances to represent, one each for Legacy and UEFI.

You can determine if the **DCIM_BootConfigSetting** represents Legacy, using the following properties:

- InstanceID = "DCIM:BootConfigSetting:Next:1"
- ElementName = "Next Boot Configuration Setting : Boot List Type 1"

You can determine if the **DCIM_BootConfigSetting** represents UEFI, using the following properties:

- InstanceID = "DCIM:BootConfigSetting:Next:2"
- ElementName = "Next Boot Configuration Setting : Boot List Type 2"

DCIM_BootSourceSetting

This class represents the boot devices or sources. The **ElementName**, **BIOSBootString**, and **StructuredBootString** properties contain a string that identifies the boot devices. For example, floppy, hard disk, CD/DVD, network, Personal Computer Memory Card International Association (PCMCIA), Battery Electric Vehicle (BEV), or USB. Based on the boot list type of the device, an instance of **DCIM_BootSourceSetting** is associated with one of the instances of **DCIM_BootConfigSetting**.

DCIM_OrderedComponent

The **DCIM_OrderedComponent** association class is used to associate instances of **DCIM_BootConfigSetting** with instances of **DCIM_BootSourceSetting** representing one of the boot list types (Legacy or UEFI) to which the boot devices belongs. The **GroupComponent** property of **DCIM_OrderedComponent** refers to the **DCIM_BootConfigSetting** instance and the **PartComponent** property refers to the **DCIM_BootSourceSetting** instance.

DCIM_Smart Attribute

For reading the smart attribute value, run the following commands:

For example:

- `Get-CimInstance -Namespace root\dcim\sysman DCIM_SmartAttributeInfo | Format-Table`
- Get-CimInstance -Namespace root\dcim\sysman DCIM_SmartAttributeInfo -Filter "AttributeID like '< Attribute ID Value >'

For setting up the custom threshold values, run the following commands:

For example:

- Get-CimInstance -Namespace root\dcim\sysman DCIM_SmartAttributeInfo -Filter "AttributeID like '<Attribute ID Value>'" | Invoke-CimMethod -MethodName "SetCustomThreshold" -Arguments @{CustomThresholdValue="<custom threshold value to be set>"}

DCIM_ThermalInformation

DCIM_ThermalInformation manages thermal configuration settings such as Thermal Mode, AAC Mode, and Fan Failure Mode.

- To query the thermal information about device, run the following command:

  ```
  Get-CimInstance -Namespace root/DCIM/SYSMAN -ClassName DCIM_ThermalInformation
  ```

- To set the value of thermal mode, run the following command:

  ```
  Get-CimInstance -Namespace root\dcim\sysman -ClassName DCIM_ThermalInformation |Where-
  Object {$_.AttributeName -eq "Thermal Mode"} | Invoke-CimMethod -MethodName
  ChangeThermalMode -Arguments @{AttributeName=@("Thermal Mode");AttributeValue=@("2")}
  ```

Changing the system settings

In Dell Command | Monitor, use the following methods to change the system settings and the state of the local or remote systems:

- SetBIOSAttributes — Changes the BIOS setting

  ⓘ **NOTE:** Dell Command | Monitor for Linux currently supports only SetBIOSAttributes method.

- ChangeBootOrder — Changes the boot configuration
- RequestStateChange — Shuts down and restarts the system
- ManageTime — Displaying the system time

In Dell Command | Monitor for Windows, you can run these methods using winrm, VB script, PowerShell commands, wmic, and WMI wbemtest.

Resetting the BIOS to defaults for systems running on Windows or Linux

ResetBIOSDefaults (Method)

PowerShell (WMI) Command: `Get-CimInstance -Namespace root\dcim\sysman -ClassName DCIM_BIOSService | Invoke-CimMethod -MethodName ResetBIOSDefaults -Arguments @{ DefaultType=<one of the possible values>}`

OMI Command: `/omicli iv root/dcim/sysman { DCIM_BIOSService Name DCIM:BiosService SystemCreationClassName DCIM_ComputerSystem SystemName <ServiceTag> CreationClassName DCIM_BIOSService } ResetBIOSDefaults { DefaultType <one of the possible values> }`

Possible values:

- 0-Built-in Safe Defaults—This is known as the BIOS defaults. This configuration supports any platforms, and therefore the configuration cannot be changed.

- 1- Last Known Good—This is generated automatically by the BIOS after successfully completing the POST. This option reverts the BIOS to a good configuration. If the **Built-in safe defaults** are corrupted, then you can use **Last Known Good** to restore the BIOS.
- 2-Factory Defaults— Factory Defaults are generated before shipping the system. This configuration is optimized for the hardware configuration or is customized as per your requirement during the purchase or service.
- 3-User-Configuration 1—This is configured when required by the user. You have to **Save Current Configuration** in the BIOS setup or F2 screen to reset the configuration through the Dell Command | Monitor application.
- 4-User-Configuration 2—This is configured when required by the user. You have to **Save Current Configuration** in the BIOS setup or F2 screen to reset the configuration through the Dell Command | Monitor application.

Table 1. ResetBIOSDefaults Possible Values

Description	Error Code (SetResult value)
Success	0
Invalid input value / Out of range input value	1
Authentication Error	2
Unsupported configuration	3
Empty configuration	4
Generic failure/Unsuccessful/when service not running	4294967295

ⓘ **NOTE:** A system restart is required post **ResetBIOSDefaults** operation for the changes to be reflected successfully.

DCM windows has the functionality to shutdown or restart the system through the following APIs:

- Restart system—`Get-CimInstance -Namespace ROOT\DCIM\SYSMAN -ClassName DCIM_ComputerSystem |Where-Object {$_.Dedicated -ne 28} | Invoke-CimMethod -MethodName RequestStateChange -Arguments @{RequestedState ='11'}`
- Shutdown system—`Get-CimInstance -Namespace ROOT\DCIM\SYSMAN -ClassName DCIM_ComputerSystem |Where-Object {$_.Dedicated -ne 28} | Invoke-CimMethod -MethodName RequestStateChange -Arguments @{RequestedState ='3'}`

The following APIs can be used in the GNU or Linux operating system:

- shutdown -r +5
- sudo reboot

ⓘ **NOTE:** During the reset operation, a subset of options is not reset to defaults. This can be either for security reasons (for example, passwords) or ability to boot (for example, Boot List and Legacy Option ROMs).

The BIOS event logs are not reset to store the history of the system hardware.

The following table enumerates the comprehensive list of features which are not reset to defaults.

Table 2. Comprehensive list of features which are not reset to defaults

Section	Subsection	Item
General	System Information	Service Tag
	System Information	Asset Tag
	System Information	Ownership Tag
	Boot Sequence	Boot List
	Advanced Boot Options	Enable Legacy OROMS
	Date/Time	Date/Time
	Integrated NIC	Integrated NIC
	Integrated NIC	Enable UEFI Network Stack
	SATA Operation	SATA Operation
Security	NA	Admin Password

Table 2. Comprehensive list of features which are not reset to defaults (continued)

Section	Subsection	Item
	NA	System Password
	NA	Internal HDD-x Password(s)
	NA	Master Password Lockout
	SMM Security Mitigation	SMM Security Mitigation
	Intel SGX Enable	Intel SGX Enable
Secure Boot	Secure Boot Enable	Secure Boot Enable
	Expert Key Management	Key Databases

Setting BIOS attributes in a system running Windows using PowerShell commands

You can set BIOS attributes using the SetBIOSAttributes method. The procedure is explained below using a task of enabling the Trusted Platform Module (TPM) as an example.

NOTE: Make sure the TPM option is cleared in the BIOS before following the procedure to enable the TPM.

NOTE: Use PowerShell with Administrator privileges.

To enable TPM,

1. Set the BIOS password on the system if not set already using the following PowerShell command:

   ```
   Get-CimInstance -Namespace root\dcim\sysman -ClassName DCIM_BIOSService
   | Invoke-CimMethod -MethodName SetBIOSAttributes -Arguments
   @{AttributeName=@("AdminPwd");AttributeValue=@("<Admin password>")}
   ```

2. Enable TPM security using the following command:

   ```
   Get-CimInstance -Namespace root\dcim\sysman -ClassName DCIM_BIOSService | Invoke-
   CimMethod -MethodName SetBIOSAttributes -Arguments @{AttributeName=@("Trusted Platform
   Module ");AttributeValue=@("1");AuthorizationToken="<Admin password>"}
   ```

3. Restart the system.
4. Activate the TPM using the following command:

   ```
   Get-CimInstance -Namespace root\dcim\sysman -ClassName DCIM_BIOSService | Invoke-
   CimMethod -MethodName SetBIOSAttributes -Arguments @{AttributeName=@(" Trusted Platform
   Module Activation");AttributeValue=@("2");AuthorizationToken="<Admin password>"}
   ```

5. Restart the system.

   Generic disclaimer

   Powershell PSReadline module saves every console command that you enter to a text file. So it is recommended to use "Get-Credential" comandlet to handle the password securely.

   a. $cred = Get-Credential

   b. Enter your username and password, for example, AdminPWD and Dell_123$, when the dialog box is displayed.

   c. $BSTR = [System.Runtime.InteropServices.Marshal]::SecureStringToBSTR($cred.Password)

   d. $plainpwd=[System.Runtime.InteropServices.Marshal]::PtrToStringAuto($BSTR)

   e. Get-CimInstance -Namespace root\dcim\sysman -ClassName DCIM_BIOSService | Invoke-CimMethod MethodName SetBIOSAttributes -Arguments @{AttributeName=@("AdminPwd");AttributeValue=@(" $plainpwd ")}

Setting BIOS attributes in the system running Linux

You can set BIOS attributes using any of the following methods:

- Using OMICLI
- Using WinRM
- Using WSMan

NOTE: Ensure that the OMI server is started and running.

Setting BIOS attributes using OMICLI

You can set BIOS attributes using the SetBIOSAttributes method. The procedure is explained below using a task of enabling the Trusted Platform Module (TPM) as an example.

NOTE: Make sure the TPM option is cleared in the BIOS before following the procedure to enable the TPM.

To set the BIOS attributes using OMICLI commands:

1. To set the BIOS password on the system if not set already, run

```
./omicli iv root/dcim/sysman { DCIM_BIOSService Name DCIM:BiosService
SystemCreationClassName DCIM_ComputerSystem SystemName <system name displayed in
DCIM_BIOSService class> CreationClassName DCIM_BIOSService } SetBIOSAttributes {
AttributeName "AdminPwd" AttributeValue “<new Admin Password>” }
```

2. To enable the TPM security use the following command, run

```
./omicli iv root/dcim/sysman { DCIM_BIOSService Name DCIM:BiosService
SystemCreationClassName DCIM_ComputerSystem SystemName <system name displayed in
DCIM_BIOSService class> CreationClassName DCIM_BIOSService } SetBIOSAttributes {
AttributeName "Trusted Platform Module" AttributeValue “1” AuthorizationToken
“<password>”
```

3. Restart the system.
4. To activate the TPM, run

```
./omicli iv root/dcim/sysman { DCIM_BIOSService Name DCIM:BiosService
SystemCreationClassName DCIM_ComputerSystem SystemName <system name displayed in
DCIM_BIOSService class> CreationClassName DCIM_BIOSService } SetBIOSAttributes
{ AttributeName " Trusted Platform Module Activation" AttributeValue “2”
AuthorizationToken “<password>” }
```

5. Restart the system.
6. To reset BIOS password, run

```
./omicli iv root/dcim/sysman { DCIM_BIOSService Name DCIM:BiosService
SystemCreationClassName DCIM_ComputerSystem SystemName <system name displayed in
DCIM_BIOSService class> CreationClassName DCIM_BIOSService } SetBIOSAttributes {
AttributeName "AdminPwd" AttributeValue “” AuthorizationToken “<password>”  }
```

Setting BIOS attributes using WinRM

You can set BIOS attributes using the SetBIOSAttributes method. The procedure is explained below using a task of enabling the Trusted Platform Module (TPM) as an example.

NOTE: Ensure the TPM option is cleared in the BIOS before following the procedure to enable the TPM.

To set the BIOS attributes using WinRM commands:

1. Get the selector set by enumerating the DCIM_BIOSService class. Run:

```
winrm e wsman/DCIM_BIOSService?__cimnamespace=root/dcim/sysman -auth:basic -r:https://
<system IP or system name>:<Port Number (5985/5986)> -username:<user name>
-password:<password> -skipCAcheck -skipCNcheck -encoding:utf-8 -returnType:epr
```

NOTE: The selector set values (SystemName=<system name from DCIM_BIOSService class>winrm i SetBIOSAttributes wsman/DCIM_BIOSService?SystemName=dt:+SystemCreationClassName=DCIM_ComputerSystem+Name=DCIM:BiosService+CreationClassName=DCIM_BIOSService+) are used for set operation in this example.

2. Set the BIOS password on the system if not set already using the following command:

```
winrm i SetBIOSAttributes http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/
DCIM_BIOSService?__cimnamespace=root/dcim/
sysman+Name=DCIM:BiosService+SystemCreationClassName=DCIM_ComputerSystem+SystemName=<syst
em name from DCIM_BIOSService class>+CreationClassName=DCIM_BIOSService -r:https://
<system IP or system name>:5986 -u:<user name> -password:<password> -auth:basic
-skipCAcheck -skipCNcheck -encoding:utf-8
@{AttributeName="AdminPwd";AttributeValue="<Password>"}
```

3. Enable TPM security by running the following command:

```
winrm i SetBIOSAttributes "http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/
DCIM_BIOSService?__cimnamespace=root/dcim/
sysman+Name=DCIM:BiosService+SystemCreationClassName=DCIM_ComputerSystem+SystemName=<syst
em name from DCIM_BIOSService class>+CreationClassName=DCIM_BIOSService -r:https://
<system IP or system name>:5986 -u:<user name> -password:<password> -auth:basic
-skipCAcheck -skipCNcheck -encoding:utf-8 @{AttributeName="Trusted Platform
Module";AttributeValue="1";AuthorizationToken="<Admin password>"}
```

4. Restart the system.
5. Activate the TPM using the following command:

```
winrm i SetBIOSAttributes "http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/
DCIM_BIOSService?__cimnamespace=root/dcim/
sysman+Name=DCIM:BiosService+SystemCreationClassName=DCIM_ComputerSystem+SystemName=<syst
em name from DCIM_BIOSService class>+CreationClassName=DCIM_BIOSService -r:https://
<system IP or system name>:5986 -u:<user name> -password:<password> -auth:basic
-skipCAcheck -skipCNcheck -encoding:utf-8 @{AttributeName=("Trusted Platform Module
Activation");AttributeValue=("2");AuthorizationToken="<Admin password>"}
```

Setting BIOS attributes using WSMan

You can set BIOS attributes on the systems running Linux using the WSMan. The procedure is explained below using a task of enabling the Trusted Platform Module (TPM) as an example.

NOTE: Ensure that the TPM option is cleared in the BIOS before following the procedure to enable the TPM.

1. Get the selector set by enumerating the DCIM_BIOSService class. Run:

```
wsman invoke –a “SetBIOSAttributes” http://
schemas.dmtf.org/wbem/wscim/1/cim-schema/2/DCIM_BIOSService?Name=”DCIM:BIOSService”,
SystemCreationClassName=”DCIM_ComputerSystem”, SystemName=”<system name from
DCIM_BIOSService class>”, CreationClassName=”DCIM_BIOSService” –N root/dcim/sysman –
h <system IP/name> –P 5985 –u <user name> –p <password> –y basic  –v  –V –k
“AttributeName=AdminPwd” –k “AttributeValue=<password>”
```

2. Set the BIOS password on the system if not set already using the following command:

```
wsman invoke –a “SetBIOSAttributes” http://
schemas.dmtf.org/wbem/wscim/1/cim-schema/2/DCIM_BIOSService?Name=”DCIM:BIOSService”,
SystemCreationClassName=”DCIM_ComputerSystem”, SystemName=”<system name from
DCIM_BIOSService class>”, CreationClassName=”DCIM_BIOSService” –N root/dcim/sysman –
h <system IP or system name> –P 5985 –u <user name> –p <password> –y basic
–v  –V –k “AttributeName=Trusted Platform Module” –k “AttributeValue=1” –k
“AuthorizationToken=<password>”
```

3. Enable TPM security using the following command:

```
wsman invoke –a “SetBIOSAttributes” http://
schemas.dmtf.org/wbem/wscim/1/cim-schema/2/DCIM_BIOSService?Name=”DCIM:BIOSService”,
SystemCreationClassName=”DCIM_ComputerSystem”, SystemName=”<system name from
```

```
DCIM_BIOSService class>”, CreationClassName=”DCIM_BIOSService” -N root/dcim/sysman -h
<system IP or system name> -P 5985 -u <user name> -p <password> -y basic  -v
-V -k “AttributeName=Trusted Platform Module Activation” -k “AttributeValue=2” -k
“AuthorizationToken=<password>”
```

4. Restart the system.
5. Activate the TPM using the following command:

```
wsman invoke -a “SetBIOSAttributes” http://
schemas.dmtf.org/wbem/wscim/1/cim-schema/2/DCIM_BIOSService?Name=”DCIM:BIOSService”,
SystemCreationClassName=”DCIM_ComputerSystem”, SystemName=”<system name from
DCIM_BIOSService class>”, CreationClassName=”DCIM_BIOSService” -N root/dcim/sysman -
h <system IP/name> -P 5985 -u <user name> -p <password> -y basic  -v  -V -k
“AttributeName=AdminPwd” -k “AttributeValue=” -k “AuthorizationToken=<password>”
```

Changing the boot sequence

To change the boot sequence follow the steps:

1. Check for the boot order type (Legacy or UEFI) by using the following command:
   - WMIC Command: `wmic /namespace:\\root\dcim\sysman path dcim_BootConfigSetting get ElementName/format:list`.
   - PowerShell Command: `Get-WmiObject -namespace root\dcim\sysman -class dcim_BootConfigSetting -Property ElementName`.
2. Check for the current boot order type (Legacy or UEFI) by using the following command:
   - WMIC Command: `wmic /namespace:\\root\dcim\sysman path dcim_ElementSettingData.IsCurrent=1 get SettingData /format:list .`
   - PowerShell Command: `Get-WmiObject -namespace root\dcim\sysman -class dcim_elementSettingData -Filter "IsCurrent=1" -Property SettingData`.
3. Changing boot-order by using the following command:
   - WMIC Command: `wmic /namespace:\\root\dcim\sysman path dcim_bootconfigsetting call ChangeBootOrder /?:full .`
   - PowerShell Command: `(Get-WmiObject -namespace root\dcim\sysman -class dcim_bootconfigsetting).getmethodparameters("ChangeBootOrder") .`

     > **NOTE:** `dcim_BootConfigSetting` instance must represent the boot configuration that you want to change – either type 1 (Legacy) or type 2 (UEFI).

   - The arguments are:
     - `Authorization Token` — This is the Administrator or boot password.
     - `Source` — This is the boot order list taken from `dcim_OrderedComponent.PartComponent` property. The new boot order is determined by the order of boot devices in the source array.
4. Changing Boot order for type 1 boot-list using PowerShell:
   a. Get Current Boot-order for type 1 boot-list by running the following command: `$boLegacy = Get-WmiObject -namespace root\dcim\sysman -class dcim_orderedcomponent | where {$_.partcomponent -match 'BootListType-1'} | select -expand partcomponent.`
   b. Define a PowerShell variable to specify boot-order to set `$newboLegacy.` Assign the new boot-order to it. For example, Current boot-order type is retained.
   c. $newboLegacy = $boLegacy
   d. Get `dcim_bootconfigsetting` instance corresponding to type 1 boot-list by running the following command: `$bcsLegacy = Get-WmiObject -Namespace root\dcim\sysman -ClassName dcim_bootconfigsetting | where {$_.ElementName -eq 'Next Boot Configuration Setting : Boot List Type 1'}.`
   e. Invoke the method by running the following command: `$ bcsLegacy.changebootorder($newboLegacy, $AuthorizationToken).` `$AuthorizationToken` variable is used to pass the BIOS password.
5. Changing Boot order for type 2 boot-list using PowerShell:
   a. Get Current Boot-order for type 2 boot-list by running the following command: `$boUefi = Get-WmiObject -namespace root\dcim\sysman -class dcim_orderedcomponent | where {$_.partcomponent -match 'BootListType-2'} | select -expand partcomponent.`

b. Define a PowerShell variable to specify boot-order to set `$newboUefi`. Assign the new boot-order to it. For example, current boot-order type is retained.

c. Get `dcim_bootconfigsetting` instance corresponding to type 2 boot-list by running the following command: `$bcsUefi = Get-WmiObject -Namespace root\dcim\sysman -ClassName dcim_bootconfigsetting | where {$_.ElementName -eq 'Next Boot Configuration Setting : Boot List Type 2'}.`

d. Invoke the method by running the following command: `$ bcsUefi.changebootorder($newboUefi, $AuthorizationToken). $AuthorizationToken` variable is used to pass the BIOS password.

Shutting down and restarting the Windows system remotely

You can shut down or restart the Windows system remotely using the RequestStateChange method.

1. Shut down the Windows system remotely using the following command:

```
(gwmi -ComputerName "SYSNAME" -Namespace root\dcim\sysman DCIM_ComputerSystem | Where-
Object {$_.Dedicated -ne 28}).RequestStateChange(3)
```

2. Restart the Windows system remotely using the following command:

```
(gwmi -ComputerName "SYSNAME" -Namespace root\dcim\sysman DCIM_ComputerSystem | Where-
Object {$_.Dedicated -ne 28}).RequestStateChange(11)
```

Getting system time value on Windows system remotely

You can get the system time value for the Windows system remotely using ManageTime method. For example:

In the command line interface, run the following:

a. `$cred = Get-Credential`

b. `$session = New-CimSession -ComputerName "Server01" -Credential $cred`

c. `Get-CimInstance -CimSession $session -Namespace root\dcim\sysman -ClassName DCIM_TimeService | Invoke-CimMethod -MethodName ManageTime -Arguments @{GetRequest="TRUE"}`

6

Managing Dell client systems locally using Dell Command | Monitor 10.6

You can manage Dell client systems locally using the following methods:

- For systems running Windows, Using PowerShell
- For systems running Linux, Using OMICLI

Topics:

Managing Windows systems locally using PowerShell

You can manage Dell client systems running Windows locally using PowerShell commands.

- Enumerating instances of DCIM class
  - `Get-CimInstance -Namespace root\dcim\sysman -ClassName DCIM_BIOSEnumeration`
  - `Get-CimInstance -Namespace root\dcim\sysman -ClassName DCIM_BIOSPassword`
- Getting properties for a BIOS setting

```
Get-CimInstance -Namespace root\dcim\sysman -ClassName DCIM_BIOSEnumeration | Where-
Object {$_.AttributeName -eq "Num Lock"}
```

- Changing BIOS settings

```
Get-CimInstance -Namespace root\dcim\sysman -ClassName DCIM_BIOSService |
Invoke-CimMethod -MethodName SetBIOSAttributes -Arguments @{AttributeName=@("Num
Lock");AttributeValue=@("1")}
```

- Modifying noncritical values

```
Get-CimInstance -Namespace root\dcim\sysman DCIM_NumericSensor | Where-Object
{$_.DeviceID -like "Root/MainSystemChassis/TemperatureObj:3"} | Set-CimInstance
-Property @{UpperThresholdNonCritical="10"}
```

- Subscribing for alerts

```
$a = 0
$timespan = New-Object System.TimeSpan(0, 0, 1)
$scope = New-Object System.Management.ManagementScope("\\.\root\dcim\sysman")
$query = New-Object System.Management.WQLEventQuery("Select * from
DCIM_AlertIndication")
$watcher = New-Object System.Management.ManagementEventWatcher($scope,$query)
[array]$alerts=@()
do{ $watcher.WaitForNextEvent() }
while ($a -ne 1)
```

- Commands to get User Consent from WMI:

```
Get-CimInstance -Namespace root\dcim\sysman -ClassName DCIM_ImprovementProgramConsent
```

- Commands to set User Consent from WMI:

```
Get-CimInstance -Namespace root\dcim\sysman -ClassName DCIM_ImprovementProgramConsent
| Invoke-CimMethod -MethodName Over
rideImprovementProgramConsent -Arguments @{NewValue="1"}
```

ⓘ **NOTE:** Improvement Program is available for Dell Command | Monitor 10.5 and 10.6 x64-bit version.

- Commands to get Proxy from WMI:

```
Get-CimInstance -Namespace root\dcim\sysman -ClassName DCIM_ApplicationProxySetting
```

- Commands to set Proxy from WMI:

```
Get-CimInstance -Namespace root\dcim\sysman -ClassName DCIM_ApplicationProxySetting |
Invoke-CimMethod -MethodName Change
ProxySetting -Arguments @{NewAddress="10.0.0.223"; NewPort="8080"}
```

Managing Linux systems locally using OMICLI

You can manage Linux systems locally using OMICLI commands. On the systems running Linux, OMICLI is installed at `/opt/omi/bin`.

- Enumerating instances of DCIM class
  - `./omicli ei root/dcim/sysman DCIM_BIOSEnumeration`
  - `./omicli ei root/dcim/sysman DCIM_BIOSPassword`
- Getting properties for a BIOS setting

```
./omicli gi root/dcim/sysman { DCIM_BIOSPassword InstanceID DCIM:BIOSSetupPassword }
```

- Setting Admin password

```
./omicli iv root/dcim/sysman { DCIM_BIOSService Name DCIM:BiosService
SystemCreationClassName DCIM_ComputerSystem SystemName <system name from
DCIM_BIOSService class> CreationClassName DCIM_BIOSService } SetBIOSAttributes {
AttributeName "AdminPwd" AttributeValue dell }
```

- Changing the BIOS settings
  - `./omicli iv root/dcim/sysman { DCIM_BIOSService Name DCIM_BiosService SystemCreationClassName DCIM_ComputerSystem SystemName <system name in DCIM_BIOSService class> CreationClassName DCIM_BIOSService } SetBIOSAttributes { AttributeName "Num Lock" AttributeValue “1” AuthorizationToken "" }`
  - `./omicli iv root/dcim/sysman { DCIM_BIOSService Name DCIM:BiosService SystemCreationClassName DCIM_ComputerSystem SystemName <system name from DCIM_BIOSService class> CreationClassName DCIM_BIOSService } SetBIOSAttributes { AttributeName "AdminPwd" AttributeValue <password> }`

```
./omicli iv root/dcim/sysman { DCIM_BIOSService Name DCIM:BiosService
SystemCreationClassName DCIM_ComputerSystem SystemName <system name from
DCIM_BIOSService class> CreationClassName DCIM_BIOSService } SetBIOSAttributes {
AttributeName "AdminPwd" AttributeValue <password> }
```

- Subscribing for alerts

```
./omicli sub root/dcim/sysman --queryexpr "select * from DCIM_AlertIndication"
```

7

Managing Dell client systems remotely using Dell Command | Monitor 10.6

You can manage Dell client systems remotely using any of the following methods:

- For systems running Windows, Managing Windows system through Windows system remotely using PowerShell on page 25
- For systems running Linux, Managing Linux system remotely through Windows system using WinRM on page 25

Topics:

Managing Windows system through Windows system remotely using PowerShell

You can access and monitor Windows system remotely through Windows system by using PowerShell.

Prerequisites for the Management Windows system:

- Windows PowerShell 3.0
- PowerShell configured for running remote scripts

Prerequisites for the Managed Windows system:

- Dell Command | Monitor
- Windows PowerShell 3.0
- PowerShell configured for running remote scripts
- PowerShell-remoting feature should be enabled

> **NOTE:**
>
> To use Windows PowerShell remotely, the remote computer must be configured for remote management. For more information, including instructions, run the PowerShell command – `Get-Help` about_remote_requirements.

Managing Linux system remotely through Windows system using WinRM

You can access and monitor the system running Linux through the system running Windows using WinRM commands.

Prerequisites for the Windows system

- Supported Windows operating system
- WinRM services running and configured for remote management

Prerequisites for the Linux system

- Root Privileges
- Dell Command | Monitor
- Supported Linux operating system
- Enable 5985 and 5986 ports on the WMI server
- System configured for your environment

In the command-line interface, run

```
winrm enumerate wsman/<DCM class name>?__cimnamespace=root/dcim/sysman -auth:basic
-r:http://<system IP or system name:5985> -username:<user name> -password:<password>
-skipCAcheck -skipCNcheck -encoding:utf-8
```

Managing Linux system remotely through a Linux system using WSMan

You can access and monitor the system running Linux remotely through the system running Linux using WSMan commands.

Prerequisites for the Management Linux system:

- Supported Linux operating system package is installed
- wsmancli package is installed

Prerequisites for the Managed Linux system:

- Root access privileges
- Supported Linux operating system
- Dell Command | Monitor

Launch a terminal, and run

```
wsman enumerate http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/root/dcim/sysman/
<class name> -N root/dcim/sysman  -h <system ip/name> -u <user name> -p <password>  -P
5985 -y basic -v -V
```

8

Frequently asked questions for Dell Command | Monitor 10.6

- How do I find the boot order (sequence) of the boot configuration using DCIM_OrderedComponent.AssignedSequence property?

  When a **DCIM_BootConfigSetting** instance (Legacy or UEFI) has multiple **DCIM_BootSourceSetting** instances (boot devices) associated with it through instances of the **DCIM_OrderedComponent** association, the value of the **DCIM_OrderedComponent.AssignedSequence** property is used to determine the sequence in which the associated **DCIM_BootSourceSetting** instances (boot devices) are used during the boot process. A **DCIM_BootSourceSetting**, whose associated **CIM_OrderedComponent.AssignedSequence** property is equal to **0** is ignored and not considered part of the boot order.

- How do I change the boot order?

  The boot order can be changed using the **DCIM_BootConfigSetting.ChangeBootOrder()** method. The **ChangeBootOrder()** method sets the order in which the instances of **DCIM_BootSourceSetting** are associated with a **DCIM_BootConfigSetting** instance. The method has one input parameter; **Source**. The **Source** parameter, is an ordered array of **PartComponent** property from **DCIM_OrderedComponent** class that represents the association between **DCIM_BootSourceSetting** instances (boot devices) and **DCIM_BootConfigSetting** instance (boot list type-Legacy or UEFI).

- How do I disable boot devices?

  On changing the boot order, the value of the **AssignedSequence** property on each instance of **DCIM_OrderedComponent**, that associates the target **DCIM_BootConfigSetting** instance with a **DCIM_BootSourceSetting** instance that is not present in the input array of **Source** parameter, is set to **0**, which indicates that the device is disabled.

- Failed login message is displayed when <what is tying to connect> tries to connect to namespace with wbemtest.

  Launch **wbemtest** with Administrator privilege level to overcome any login message. Go to the Internet Explorer from the **All Programs** list, right-click, and **Run as administrator** to start the **wbemtest** and avoid a namespace error.

- How do I run Knowledge Library scripts without any issues?

  The following are the steps to run the VBS scripts provided in Dell Command | Monitor Knowledge Library link:

  1. Configure **winrm** on the system using the command `winrm quickconfig`.
  2. Check if the token support exists on the system by seeing:
     - The **F2 Screen** in BIOS Setup.
     - Using tool like wbemtest to check that the key values define in the script to be existing on the system.

     NOTE: Dell recommends using the latest BIOS available at **dell.com/support**. For more information, see Dell Command | Monitor Reference guide at **dell.com/support**.

- How do I set the BIOS attributes?

  BIOS Attributes can be changed using the **DCIM_BIOSService.SetBIOSAttributes()** method. The **SetBIOSAttributes()** method sets the value of the instance that is defined in the **DCIM_BIOSEnumeration** class. The method has seven input parameters. The first two parameters can be empty or null. The third parameter **AttributeName** must take the input mapping to the value of attribute name instance of **DCIM_BIOSEnumeration** class. The fourth parameter or **AttributeValue** can be any of the possible values of the Attribute Name as defined in the**DCIM_BIOSEnumeration** class. The fifth parameter AuthorizationToken is optional, the input for fifth parameter is BIOS Password. The fifth parameter is used only when the BIOS Password is set on the system else it is empty. The sixth and seventh argument can again be empty or null.

- Does Dell Command | Monitor support storage and sensor monitoring for Windows and Linux operating systems?

  Yes, Dell Command | Monitor supports both storage and sensor monitoring for supported Windows and Linux operating systems.

  In storage monitoring, Dell Command | Monitor supports monitoring and alerting of:

- Intel-integrated controller (compliant with CSMI v0.81 or later)
    - LSI-integrated RAID controllers; and 9217, 9271, 9341, 9361 and their associated drivers(physical and logical)

    ⓘ **NOTE:** Monitoring of Intel-integrated controller is not supported for the systems running Linux operating system.

    In sensor monitoring, Dell Command | Monitor supports monitoring and alerting of voltage, temperature, amperage, cooling devices (fan) and chassis sensors.

    For more information about class and alerting, see Dell Command | Monitor Reference guide at **dell.com/support**.

- Can Dell Command | Monitor be integrated with other applications/consoles?

  Yes, Dell Command | Monitor interfaces with leading enterprise management console that support industry standards. It can be integrated with the following existing enterprise management tools:

    - Dell Client Integration Suite for System Center 2012
    - Dell OpenManage Essentials
    - Dell Client Management Pack for System Center Operation Manager
- Can I import classes into SCCM for inventory?

  Yes, individual MOFs or OMCI_SMS_DEF.mof files can be imported in SCCM console for inventory.

- Where is the SCCM OMCI_SMS_DEF.mof file located?

  The OMCI_SMS_DEF.mof file is at C:\Program Files\Dell\Command_Monitor\ssa\omacim\OMCI_SMS_DEF.mof.

- How to configure proxy for DCM 10.2.1?

- DCM 10.2.1 is unable to fetch warranty information.

- Check if the Application proxy settings are correctly configured using DCIM_ApplicationProxySetting Class.

  How can I configure a Proxy credential for Dell Command | Monitor.

  If you have logged in through Dell Command | Monitor, you can use the same credentials for proxy authentication.

- Dell Command | Monitor is not displaying the warranty information.

    - Client system is not connected to the internet during polling.

      Connect the internet and pull warranty information by running the following command `Get-CimInstance -Namespace root/DCIM/SYSMAN -ClassName DCIM_AssetWarrantyInformation| Where-Object{$_.InstanceID -eq "Root/MainSystemChassis/COOObject/COOWarranty:0"} | Invoke-CimMethod -MethodName RefreshWarranty`

    - Client system is not configured with the proxy server.

      Configure proxy settings in Dell Command | Monitor by running the following commands:

        - To get proxy from the WMI, run the following command `Get-CimInstance -Namespace root\dcim\sysman -ClassName DCIM_ApplicationProxySetting`.
        - To set proxy from the WMI, run the following command `Get-CimInstance -Namespace root\dcim\sysman -ClassName DCIM_ApplicationProxySetting | Invoke-CimMethod -MethodName Change ProxySetting -Arguments @{NewAddress="10.0.0.223"; NewPort="8080"}`

          You have to replace the **NewAddress** and **NewPort** as per the proxy environment (if applicable).

9

Troubleshooting steps using Dell Command | Monitor 10.6

Topics:

Unable to remotely connect to Windows Management Instrumentation

If Common Information Model (CIM) information for a remote client computer system is not available to the management application or if a remote BIOS update that uses Distributed Component Object Model (DCOM) fails, the following error messages are displayed:

- **Access Denied**
- **Win32:RPC server is unavailable**

1. Verify that the client system is connected to the network. Type the following in the command prompt of the server: `ping <Host Name or IP Address>` and press `<Enter>`.
2. Perform the following step if both the server and the client system are in the same domain:
   - Verify that the domain administrator account has Administrator privileges for both systems.

   Perform the following step if both the server and the client system are in a workgroup (not in the same domain):
   - Ensure that the server is running on the latest Windows Server.

   NOTE: Back up your system data files before changing the registry. Editing the registry incorrectly may render your operating system unusable.

3. Edit the registry change on the client system. Click **Start** > **Run**, then type **regedit**, and then click **OK**. In the **Registry Editor** window, browse to `My Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa`.
4. Set the **forceguest** value to 0 (default value is 1). Unless you modify this value, the user remotely connecting to the system has guest privileges, even if the supplied credentials provide Administrator privileges.
   a. Create an account on the client system with the same username and password, as an administrator account on the system running the WMI management application.
   b. If you are using IT Assistant, run the IT Assistant ConfigServices utility (`configservices.exe` in the`/bin` directory under the IT Assistant installation directory). Configure IT Assistant to run under a local administrator account, which is also now an administrator on the remote client. Also, verify that DCOM and CIM are enabled.
   c. If you are using IT Assistant, use the administrator account to configure subnet discovery for the client system. Enter the username as <client machine name>\<account name>. If the system has already been discovered, remove the system from the list of discovered systems, configure subnet discovery for it, and then rediscover it.

   NOTE: Dell recommends using Dell OpenManage Essentials as replacement for IT Assistant. For more information on Dell OpenManage Essentials, see, **dell.com/support**.

5. Perform the following steps to modify user privilege levels for connecting remotely to a system's WMI:
   a. Click **Start** > **Run**, type `compmgmt.msc`, and then click **OK**.
   b. Browse to **WMI Control** under **Services and Applications**.
   c. Right-click **WMI Control**, and then click **Properties**.

   d. Click the **Security** tab and select **DCIM/SYSMAN** under the **Root** tree.
   e. Click **Security**.
   f. Select the specific group or user that you want to control access and use the **Allow** or **Deny** check box to configure the permissions.
6. Perform the following steps to connect to a WMI (`root\DCIM\SYSMAN`) on a system from a remote system using WMI CIM Studio:
   a. Install WMI tools along with wbemtest on the local system, and then install Dell Command | Monitor on the remote system.
   b. Configure the firewall on the system for WMI remote connectivity. For example, open the TCP ports 135 and 445 in Windows firewall.
   c. Set the Local Security setting to **Classic - local users authenticate as themselves for Network access: Sharing and security model for local accounts** in the **Local Security Policy**.
   d. Connect to the WMI (`root\DCIM\SYSMAN`) on the local system from a remote system using WMI wbemtest. For example, \\[Target remote system IP Address]\root\DCIM\SYSMAN
   e. Enter the Administrator credentials of the target remote system if prompted.

   For more information about WMI, see the applicable Microsoft documentation at msdn.microsoft.com.

Installation failure on systems running Windows

If you are unable to complete Dell Command | Monitor for Windows installation, ensure that:

- You have Administrator privileges on the target system.
- The target system is a Dell manufactured system with SMBIOS version 2.3 or later.
- PowerShell console must not be open.

NOTE: To check the SMBIOS version on the system, go to **Start** > **Run** and run the `msinfo32.exe` file and check for the SMBIOS version in System Summary page.

NOTE: The system must be running supported Windows operating system.

NOTE: The system has to be upgraded to .NET 4.0 or later versions.

BIOS setting enumeration value appears as 1

1. Verify that the following packages are installed with root user privileges;
   - omi-1.0.8.ssl_100.x64.rpm
   - srvadmin-hapi-8.3.0-1908.9058.el7.x86_64
   - command_monitor-linux-<version number>-<buid number>.x86_64.rpm
2. If above packages are installed, then verify that the driver module is loaded.
   a. Verify that the driver module is loaded by running the following command `lsmod | grep dcdbas`.
   b. If the driver module is not available, retrieve the driver details by running the following command `modinfo dcdbus`.
   c. Load the driver module by running the following command `insmod <filename>`.

Hapi installation fails due to the dependency of libsmbios

If the installation fails due to dependency problems,

Force-install all dependent packages by running `apt-get -f install`.

CIM resources not available

While enumerating, if you receive an error as “CIM resource not available”,

Verify that the commands are executed with root privileges.

Unable to execute the commands using DCM on the systems running Ubuntu Core 16

Ensure that the snap version on the system is 2.23 or later.

Other documents you may need

In addition to this User's Guide, you can access the following documents at **dell.com/support**. Click Dell Command | Monitor (formerly OpenManage Client Instrumentation) and then click the appropriate product version link in **General support** section.

In addition to this User's Guide, you can access the following guides.

- The Dell Command | Monitor Reference Guide provides detailed information on all classes, properties, and descriptions.
- The Dell Command | Monitor Installation Guide provides information on installation.
- The Dell Command | Monitor SNMP Reference Guide provides Simple Network Management Protocol (SNMP) Management Information Base (MIB) applicable to Dell Command | Monitor.

Topics:

Accessing documents from the Dell support site

You can access the required documents by selecting your product.

1. Go to www.dell.com/manuals.
2. Click **Browse all products**, click **Software**, and then click **Client Systems Management**.
3. To view the required documents, click the required product name and version number.

11

Contacting Dell

NOTE: If you do not have an active Internet connection, you can find contact information on your purchase invoice, packing slip, bill, or Dell product catalog.

Dell provides several online and telephone-based support and service options. Availability varies by country and product, and some services may not be available in your area. To contact Dell for sales, technical support, or customer service issues:

1. Go to **Dell.com/support**.
2. Select your support category.
3. Verify your country or region in the **Choose a Country/Region** drop-down list at the bottom of the page.
4. Select the appropriate service or support link based on your need.