|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | | |  |
|  | | |
|  | | |
|
|  | **xFusion-iBMC-Cmdlets**  **V1.3.6**  **User Guide** | |  |  |
|  | |
| **Date** | **2022-01-30** |
| Copyright © Copyright Holder. All rights reserved. | |

|  |
| --- |
| Notice  The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied. |

Contents

[About This Document vii](#_Toc96528720)

[1 Introduction 9](#_Toc96528721)

[2 Installing and Uninstalling xFusion-iBMC-Cmdlets 11](#_Toc96528722)

[2.1 Preparations 11](#_Toc96528723)

[2.2 Installing and Uninstalling xFusion-iBMC-Cmdlets 13](#_Toc96528724)

[3 Using xFusion-iBMC-Cmdlets 15](#_Toc96528725)

[3.1 Viewing the Help Information 22](#_Toc96528726)

[3.2 Querying the xFusion-iBMC-Cmdlets Version Information 25](#_Toc96528727)

[3.3 Connecting to the iBMC 26](#_Toc96528728)

[3.4 Disconnecting from the iBMC 30](#_Toc96528729)

[3.5 Testing the iBMC Connection Status 31](#_Toc96528730)

[3.6 Querying the Session Service Information 33](#_Toc96528731)

[3.7 Modifying the Session Service Information 34](#_Toc96528732)

[3.8 Querying the User List 35](#_Toc96528733)

[3.9 Creating a User 36](#_Toc96528734)

[3.10 Modifying a User 38](#_Toc96528735)

[3.11 Removing a User 39](#_Toc96528736)

[3.12 Querying the iBMC Service Information 40](#_Toc96528737)

[3.13 Setting the iBMC Service Information 41](#_Toc96528738)

[3.14 Querying the Power Consumption Information 42](#_Toc96528739)

[3.15 Querying the System Information 43](#_Toc96528740)

[3.16 Querying the Ethernet Port Information 46](#_Toc96528741)

[3.17 Setting the Power Supply Status 47](#_Toc96528742)

[3.18 Restarting the iBMC 48](#_Toc96528743)

[3.19 Restarting the System 49](#_Toc96528744)

[3.20 Querying the Asset Tag 50](#_Toc96528745)

[3.21 Setting the Asset Tag 50](#_Toc96528746)

[3.22 Querying the Virtual Media Information 51](#_Toc96528747)

[3.23 Connecting to the Virtual Media 52](#_Toc96528748)

[3.24 Disconnecting from the Virtual Media 54](#_Toc96528749)

[3.25 Querying the Boot Device 55](#_Toc96528750)

[3.26 Setting the Boot Device 56](#_Toc96528751)

[3.27 Querying the Boot Sequence 57](#_Toc96528752)

[3.28 Setting the Boot Sequence 58](#_Toc96528753)

[3.29 Exporting the BIOS Configuration 59](#_Toc96528754)

[3.30 Importing the BIOS Configuration 61](#_Toc96528755)

[3.31 Restoring the Factory Settings 63](#_Toc96528756)

[3.32 Restoring the Default BIOS Settings 64](#_Toc96528757)

[3.33 Querying BIOS Settings 65](#_Toc96528758)

[3.34 Modifying BIOS Settings 66](#_Toc96528759)

[3.35 Querying the NTP Resource Information 67](#_Toc96528760)

[3.36 Setting NTP Resource Attributes 68](#_Toc96528761)

[3.37 Importing the NTP Group Key 70](#_Toc96528762)

[3.38 Querying the SMTP Notification List 71](#_Toc96528763)

[3.39 Setting the SMTP Notification Object 73](#_Toc96528764)

[3.40 Querying the SMTP Resource Information 74](#_Toc96528765)

[3.41 Setting SMTP Resource Attributes 75](#_Toc96528766)

[3.42 Querying the SNMP Resource Information 77](#_Toc96528767)

[3.43 Setting SNMP Resource Attributes 78](#_Toc96528768)

[3.44 Querying the SNMP Trap Server 80](#_Toc96528769)

[3.45 Setting the SNMP Trap Server 82](#_Toc96528770)

[3.46 Querying the SNMP Trap Resource Information 83](#_Toc96528771)

[3.47 Setting SNMP Trap Resource Attributes 84](#_Toc96528772)

[3.48 Querying the Syslog Notification List 86](#_Toc96528773)

[3.49 Setting the Syslog Notification Object 87](#_Toc96528774)

[3.50 Querying the Syslog Resource Information 88](#_Toc96528775)

[3.51 Setting the Syslog Resource Information 89](#_Toc96528776)

[3.52 Querying the In-Band Firmware Version 91](#_Toc96528777)

[3.53 Querying the Out-of-Band Firmware Version 91](#_Toc96528778)

[3.54 Upgrading the In-Band Firmware 92](#_Toc96528779)

[3.55 Upgrading the Out-of-Band Firmware 95](#_Toc96528780)

[3.56 Querying the CPU Health Status 97](#_Toc96528781)

[3.57 Querying the RAID Controller Card Health Status 98](#_Toc96528782)

[3.58 Querying the Memory Health Status 99](#_Toc96528783)

[3.59 Querying the Power Supply Health Status 100](#_Toc96528784)

[3.60 Querying the Drive Health Status 101](#_Toc96528785)

[3.61 Querying the NIC Health Status 102](#_Toc96528786)

[3.62 Querying the Fan Health Status 103](#_Toc96528787)

[3.63 Querying the CPU Information 104](#_Toc96528788)

[3.64 Querying the RAID Controller Card Information 107](#_Toc96528789)

[3.65 Querying the Memory Information 110](#_Toc96528790)

[3.66 Querying the Power Supply Information 112](#_Toc96528791)

[3.67 Querying the Drive Information 115](#_Toc96528792)

[3.68 Querying the NIC Information 121](#_Toc96528793)

[3.69 Querying the Fan Information 123](#_Toc96528794)

[3.70 Querying the Configuration Result Resource of the Smart Provisioning Service 125](#_Toc96528795)

[3.71 Modifying the Smart Provisioning Service Resource Attributes 127](#_Toc96528796)

[3.72 Exporting the RAID Configuration of the Smart Provisioning Service 127](#_Toc96528797)

[3.73 Creating the RAID Configuration of the Smart Provisioning Service 128](#_Toc96528798)

[3.74 Deleting the RAID Configuration of the Smart Provisioning Service 130](#_Toc96528799)

[3.75 Modifying the RAID Configuration of the Smart Provisioning Service 131](#_Toc96528800)

[3.76 Querying the RAID Configuration Resource of the Smart Provisioning Service 133](#_Toc96528801)

[3.77 Querying the OS Installation Configuration Resources of the Smart Provisioning Service 134](#_Toc96528802)

[3.78 Creating the OS Installation Configuration for the Smart Provisioning Service 136](#_Toc96528803)

[3.79 Restoring the Default Configuration of a Specified RAID Controller Card 143](#_Toc96528804)

[3.80 Modifying the Information of a Specified RAID Controller Card Resource 143](#_Toc96528805)

[3.81 Modifying the Attributes of a Specified Drive 144](#_Toc96528806)

[3.82 Querying the Logical Drive Resource Information 146](#_Toc96528807)

[3.83 Modifying the Logical Drive Resource Attributes 149](#_Toc96528808)

[3.84 Initializing a Logical Drive 152](#_Toc96528809)

[3.85 Deleting a Logical Drive 153](#_Toc96528810)

[3.86 Creating a Logical Drive 154](#_Toc96528811)

[3.87 Collecting iBMC Logs in One-click Mode 158](#_Toc96528812)

[3.88 Downloading a BMC File 161](#_Toc96528813)

[3.89 Uploading a File 162](#_Toc96528814)

[3.90 Querying the License Service Information 163](#_Toc96528815)

[3.91 Installing a License 165](#_Toc96528816)

[3.92 Exporting a License 167](#_Toc96528817)

[3.93 Deleting a License 168](#_Toc96528818)

[3.94 Querying iBMC Network Port Information 169](#_Toc96528819)

[3.95 Setting iBMC Network port Information 171](#_Toc96528820)

[3.96 Querying LDAP Information 172](#_Toc96528821)

[3.97 Modifying LDAP Information 173](#_Toc96528822)

[3.98 Importing an LDAP certificate to the iBMC 175](#_Toc96528823)

[3.99 Setting the LDAP Status 178](#_Toc96528824)

[3.100 Querying Kerberos Service Resources 179](#_Toc96528825)

[3.101 Changing the Kerberos Enabling Status 180](#_Toc96528826)

[3.102 Querying the Kerberos Domain Controller Information 181](#_Toc96528827)

[3.103 Modifying the Kerberos Domain Controller Information 183](#_Toc96528828)

[3.104 Importing a Kerberos Domain Controller Key Table 186](#_Toc96528829)

[3.105 Querying a Diagnostic Service Resource 187](#_Toc96528830)

[3.106 Modifying Diagnostic Service Resource Information 189](#_Toc96528831)

[3.107 General Interface 190](#_Toc96528832)

[3.108 Importing the Root Certificate of a Remote HTTPS Server 191](#_Toc96528833)

[3.109 Deleting the Root Certificate of a Remote HTTPS Server 194](#_Toc96528834)

[3.110 Importing a CRL of a Remote HTTPS Server 195](#_Toc96528835)

[3.111 Querying the Certificate Verification Switch Status of the Remote HTTPS Server 198](#_Toc96528836)

[3.112 Enabling/Disabling the Certificate Verification of the Remote HTTPS Server 199](#_Toc96528837)

[4 FAQs 201](#_Toc96528838)

[4.1 How to Use an Encrypted Password When Invoking Cmdlets 201](#_Toc96528839)

[4.2 How to Use the CA Root Certificate to Authenticate a Server 202](#_Toc96528840)

[4.3 How to Invoke a Cmdlets Interface 211](#_Toc96528841)

[4.4 How to Record Logs as a Non-Administrator User 212](#_Toc96528842)

[A Public IP Addresses and Email Addresses 214](#_Toc96528843)

[B Obtaining Technical Support 215](#_Toc96528844)

About This Document

Purpose

This document describes how to install and use xFusion-iBMC-Cmdlets.

Intended Audience

This document is intended for:

* Technical support engineers
* Maintenance engineers

Symbol Conventions

The symbols that may be found in this document are defined as follows.

| Symbol | Description |
| --- | --- |
|  | Indicates an imminently hazardous situation which, if not avoided, will result in death or serious injury. |
|  | Indicates a potentially hazardous situation which, if not avoided, could result in death or serious injury. |
|  | Indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury. |
|  | Indicates a potentially hazardous situation which, if not avoided, could result in equipment damage, data loss, performance deterioration, or unanticipated results.  NOTICE is used to address practices not related to personal injury. |
|  | Calls attention to important information, best practices and tips.  NOTE is used to address information not related to personal injury, equipment damage, and environment deterioration. |

Change History

Changes between document issues are cumulative. The latest document issue contains all the changes made in earlier issues.

| Issue | Date | Description |
| --- | --- | --- |
| 01 | 2022-01-30 | This issue is the first official release. |

# Introduction

Function

xFusion-iBMC-Cmdlets is a command-line tool developed based on the PowerShell language. It connects to the iBMC over the Redfish interface to implement management of servers. It supports the following functions:

* Query of basic server information
* Configuration of iBMC and BIOS parameters
* Upgrade of the iBMC, BIOS, and CPLD firmware
* Upgrade of the Smart Provisioning, RAID controller card, and NIC firmware
* Reset and power control of the server
* Configuration of settings related to operating system (OS) deployment
* Query of basic information and health status of components
* RAID configuration
* Log collection
* Concurrent access to a maximum of 256 servers
* Access to the iBMC using an IPv4 or IPv6 address or a domain name
* Installation, exporting, and deleting of the license
* Configuration of the Kerberos domain controller
* Query and modifying of the diagnostic service resource
* Providing of the common public interface
* Management of the root certificate of the remote HTTPS server



The Cmdlets plugin does not collect or process users' personal data.

Servers Supported by Cmdlets Plug-in

| Architecture | Type | Server Model |
| --- | --- | --- |
| x86 | Rack server | RH1288 V3 |
| RH2288 V3 |
| RH2288H V3 |
| 1288H V5 |
| 2288 V5 |
| 2288H V5 |
| 2488H V5 |
| 1288H V6 |
| 2288H V6 |
| 5288 V6 |
| Blade server | CH121 V3 |
| CH242 V3 |
| High-density server | XH622 V3 |
| XH321 V5 |

Supported Components

| Type | Model |
| --- | --- |
| RAID controller card | LSI SAS3008, LSI SAS3108, Avago SAS3408iMR, Avago SAS3508, and Avago SAS3004 |
| NIC | X722, SP212, SP330, and SP380 |

Matching Versions

| Software | Matching Versions |
| --- | --- |
| BIOS | 0.39 or later |
| iBMC | * V6 server:V3.01.12.23 or later * V5 server: V322 or later * V3 server: V308 or later |
| Smart Provisioning | V116 or later |

# Installing and Uninstalling xFusion-iBMC-Cmdlets

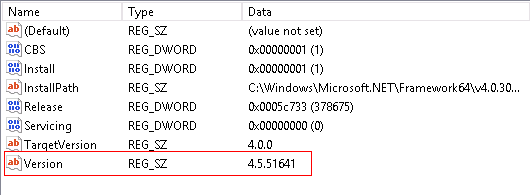
[2.1 Preparations](#_EN-US_TOPIC_0000001137285249)

[2.2 Installing and Uninstalling xFusion-iBMC-Cmdlets](#_EN-US_TOPIC_0000001137285245)

## Preparations

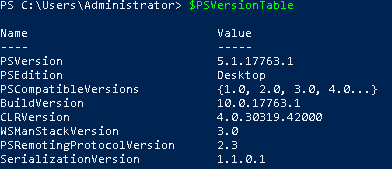
* xFusion-iBMC-Cmdlets supports .Net Framework 4.5 or later. Perform the following operations to view the .Net Framework version:
  1. In the **Start** search box, enter **regedit.exe** and press **Enter** to open the Registry Editor.
  2. The **HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\NET Framework Setup\NDP\v4\Full** file shows the .Net Framework version, as shown in Figure 2-1.

Registry Editor



* xFusion-iBMC-Cmdlets supports PowerShell 5.0 or later. Perform the following operations to view the PowerShell version:
  1. Open the PowerShell CLI.
  2. Run the **$PSVersionTable** command to view the PowerShell version, as shown in Figure 2-2.

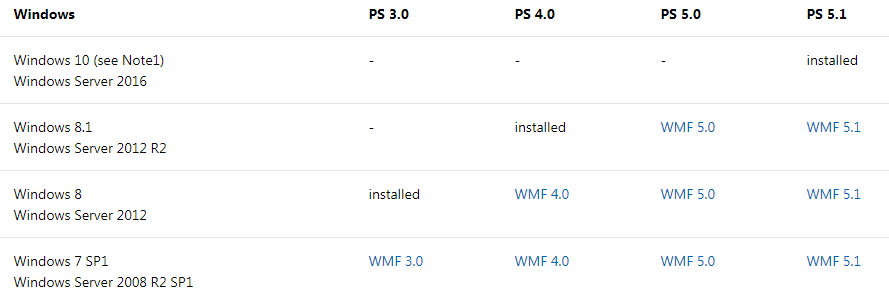
Viewing the PowerShell version



* 1. If the PowerShell version is earlier than 5.0, perform the following operations to install PowerShell 5.0 or later:

Figure 2-3 lists the PowerShell installed by default for each Windows version. PowerShell 5.1 is used as an example.

PowerShell installed by default

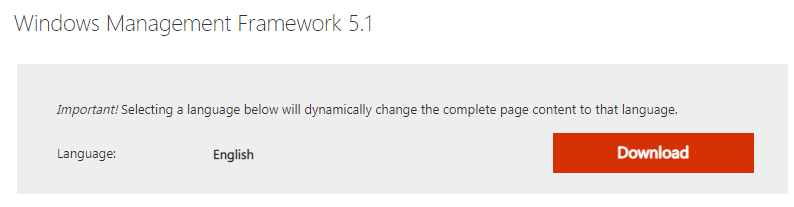


* + 1. Uninstall the PowerShell of the earlier version.
    2. Download PowerShell 5.1 from the following link:

<https://docs.microsoft.com/en-us/powershell/scripting/install/installing-windows-powershell?view=powershell-6>

* + 1. Click **Download**, select **W2K12-KB3191565-x64.msu**, and click **Next**.

PowerShell 5.1 download page



* + 1. Double-click **W2K12-KB3191565-x64.msu** and install PowerShell 5.1 using the default settings.
    2. Restart the OS for the installation to take effect.

## Installing and Uninstalling xFusion-iBMC-Cmdlets

Prerequisites

You have obtained the Cmdlets plug-in software package and verified its integrity.

1. Obtain the Cmdlets plug-in software package (for example, **xFusion-iBMC-Cmdlets v1.3.6.zip**) and its SHA256 verification file (for example, **xFusion-iBMC-Cmdlets v1.3.6.sha256.sum**) from [GitHub](https://github.com/Open-xFusion/Server_Plugin_iBMC-Cmdlets/tree/master/releases).
2. Verify the integrity of the Cmdlets plug-in software package.

* Windows OS:
  + 1. Open the CMD and go to the directory where the plug-in software package is stored.
    2. Run the **certutil -hashfile** *"software package name"* **sha256** command to check the SHA256 hash value of the software package.

Example: **certutil -hashfile "xFusion-iBMC-Cmdlets v1.3.6.zip" sha256**

* + 1. Check whether the SHA256 hash value of the software package is the same as that of the SHA256 verification file.
    - If yes, the software package has not been tampered with and can be used.
    - If no, the software package has been tampered with. Obtain a new software package.
* Linux OS:
  + 1. Go to the directory where the plug-in software package and SHA256 verification file are stored.
    2. Run the **sha256sum -c < (grep** *software package name* *sha256 verification file name***)** command to verify the software package.

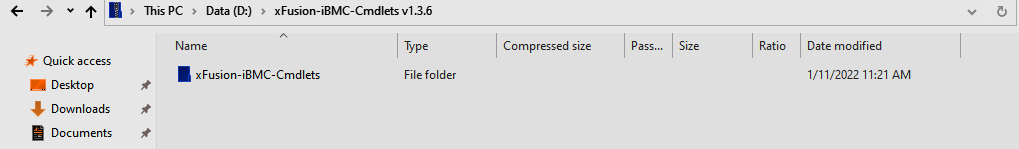
Example: **sha256sum -c <(grep xFusion-iBMC-Cmdlets v1.3.6.zip xFusion-iBMC-Cmdlets v1.3.6.sha256.sum)**

* + 1. Check whether the verification result is **OK**.
    - If yes, the software package has not been tampered with and can be used.
    - If no, the software package has been tampered with. Obtain a new software package.

Procedure

* Install xFusion-iBMC-Cmdlets.
  1. Decompress the **xFusion-iBMC-Cmdlets v1.3.6.zip** to extract the **xFusion-iBMC-Cmdlets** folder, as shown in Figure 2-6.

The **xFusion-iBMC-Cmdlets** folder



* 1. Run the **$env:PSModulePath** command to view the installation directories supported by PowerShell.

PS C:\Users\Administrator> **$env:PSModulePath**   
C:\Users\Administrator\Documents\WindowsPowerShell\Modules;C:\Program Files\WindowsPowerShell\Modules;C:\Windows\system32\WindowsPowerShell\v1.3.1\Modules

* 1. Copy the **xFusion-iBMC-Cmdlets** folder to the PowerShell installation directory.
  2. Open the PowerShell CLI.
  3. Run the following command to install xFusion-iBMC-Cmdlets:

PS C:\Users\Administrator> **Import-Module xFusion-iBMC-Cmdlets -Force**

* 1. Run the following command to check whether xFusion-iBMC-Cmdlets is successfully installed:

PS C:\Users\Administrator> **Get-Module | ? Name -eq xFusion-iBMC-Cmdlets**   
   
ModuleType Version Name ExportedCommands   
---------- ------- ---- ----------------   
Script 1.3.6 xFusion-iBMC-Cmdlets {Add-iBMCSPRAIDVolume, Add-iBMCUser, Add-iBMCVolume, Clear-iBMCSPRAIDSetting...}

* Uninstall xFusion-iBMC-Cmdlets.
  1. Open the PowerShell CLI.
  2. Run the following command to uninstall xFusion-iBMC-Cmdlets:

PS C:\Users\Administrator> **remove-module –name xFusion-iBMC-Cmdlets**

* 1. Delete the xFusion-iBMC-Cmdlets file from the PowerShell module directory.

# Using xFusion-iBMC-Cmdlets

You can run the following command to view all xFusion-iBMC-Cmdlets commands:

**Get-Command -Module xFusion-iBMC-Cmdlets**

PS C:\Users\Administrator> Get-Command -Module xFusion-iBMC-Cmdlets   
   
CommandType Name Version Source   
----------- ---- ------- ------   
Function Add-iBMCSPRAIDVolume 1.3.6 xFusion-iBMC-Cmdlets   
Function Add-iBMCUser 1.3.6 xFusion-iBMC-Cmdlets   
Function Add-iBMCVolume 1.3.6 xFusion-iBMC-Cmdlets   
Function Clear-iBMCSPRAIDSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Connect-iBMC 1.3.6 xFusion-iBMC-Cmdlets   
Function Connect-iBMCVirtualMedia 1.3.6 xFusion-iBMC-Cmdlets   
Function Disconnect-iBMC 1.3.6 xFusion-iBMC-Cmdlets   
Function Disconnect-iBMCVirtualMedia 1.3.6 xFusion-iBMC-Cmdlets   
Function Export-iBMCBIOSSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Export-iBMCLicense 1.3.6 xFusion-iBMC-Cmdlets   
Function Export-iBMCMaintenanceInfo 1.3.6 xFusion-iBMC-Cmdlets   
Function Export-iBMCSPRAIDSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCAssetTag 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCBIOSSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCBootSourceOverride 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCBootupSequence 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCDiagnosticService 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCDrives 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCDrivesHealth 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCFans 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCFansHealth 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCInbandFirmware 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCIP 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCKerberos 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCKerberosController 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCLDAP 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCLicense 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCMemory 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCMemoryHealth 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCModuleVersion 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCNetworkAdapters 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCNetworkAdaptersHealth 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCNTPSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCOSDeployConfig 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCOutbandFirmware 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCPowerInfo 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCPowerSupplies 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCPowerSuppliesHealth 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCProcessors 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCProcessorsHealth 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCRAIDControllers 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCRAIDControllersHealth 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCSecurityService 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCServices 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCSessionTimeout 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCSMTPRecipients 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCSMTPSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCSNMPSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCSNMPTrapServer 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCSNMPTrapSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCSPRAIDSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCSPTaskResult 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCSyslogServer 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCSyslogSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCSystemInfo 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCSystemNetworkSettings 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCUser 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCVirtualMedia 1.3.6 xFusion-iBMC-Cmdlets   
Function Get-iBMCVolume 1.3.6 xFusion-iBMC-Cmdlets   
Function Import-iBMCBIOSSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Import-iBMCKerberosControllerKeyTable 1.3.6 xFusion-iBMC-Cmdlets   
Function Import-iBMCLDAPCert 1.3.6 xFusion-iBMC-Cmdlets   
Function Import-iBMCNTPGroupKey 1.3.6 xFusion-iBMC-Cmdlets   
Function Import-iBMCRemoteHttpsServerCrl 1.3.6 xFusion-iBMC-Cmdlets   
Function Import-iBMCRemoteHttpsServerRootCA 1.3.6 xFusion-iBMC-Cmdlets   
Function Initialize-iBMCVolume 1.3.6 xFusion-iBMC-Cmdlets   
Function Install-iBMCLicense 1.3.6 xFusion-iBMC-Cmdlets   
Function Invoke-iBMCFileDownload 1.3.6 xFusion-iBMC-Cmdlets   
Function Invoke-iBMCFileUpload 1.3.6 xFusion-iBMC-Cmdlets   
Function Invoke-iBMCGeneralCall 1.3.6 xFusion-iBMC-Cmdlets   
Function Remove-iBMCLicense 1.3.6 xFusion-iBMC-Cmdlets   
Function Remove-iBMCRemoteHttpsServerRootCA 1.3.6 xFusion-iBMC-Cmdlets   
Function Remove-iBMCUser 1.3.6 xFusion-iBMC-Cmdlets   
Function Remove-iBMCVolume 1.3.6 xFusion-iBMC-Cmdlets   
Function Reset-iBMC 1.3.6 xFusion-iBMC-Cmdlets   
Function Reset-iBMCBIOSSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Restore-iBMCFactorySetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Restore-iBMCRAIDController 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCAssetTag 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCBIOSSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCBootSourceOverride 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCBootupSequence 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCDiagnosticService 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCDrive 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCFruControl 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCIP 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCKerberos 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCKerberosController 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCLDAP 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCLDAPServiceEnabled 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCNTPSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCOSDeployConfig 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCRAIDController 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCSecurityService 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCServerPower 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCService 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCSessionTimeout 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCSMTPRecipient 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCSMTPSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCSNMPSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCSNMPTrapServer 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCSNMPTrapSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCSPRAIDSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCSPService 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCSyslogServer 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCSyslogSetting 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCUser 1.3.6 xFusion-iBMC-Cmdlets   
Function Set-iBMCVolume 1.3.6 xFusion-iBMC-Cmdlets   
Function Test-iBMCConnect 1.3.6 xFusion-iBMC-Cmdlets   
Function Update-iBMCInbandFirmware 1.3.6 xFusion-iBMC-Cmdlets   
Function Update-iBMCOutbandFirmware 1.3.6 xFusion-iBMC-Cmdlets

[3.1 Viewing the Help Information](#_EN-US_TOPIC_0000001137285243)

[3.2 Querying the xFusion-iBMC-Cmdlets Version Information](#_EN-US_TOPIC_0000001137285237)

[3.3 Connecting to the iBMC](#_EN-US_TOPIC_0000001137285235)

[3.4 Disconnecting from the iBMC](#_EN-US_TOPIC_0000001137285233)

[3.5 Testing the iBMC Connection Status](#_EN-US_TOPIC_0000001137285231)

[3.6 Querying the Session Service Information](#_EN-US_TOPIC_0000001137285171)

[3.7 Modifying the Session Service Information](#_EN-US_TOPIC_0000001137284999)

[3.8 Querying the User List](#_EN-US_TOPIC_0000001137285043)

[3.9 Creating a User](#_EN-US_TOPIC_0000001137285085)

[3.10 Modifying a User](#_EN-US_TOPIC_0000001137285129)

[3.11 Removing a User](#_EN-US_TOPIC_0000001137285173)

[3.12 Querying the iBMC Service Information](#_EN-US_TOPIC_0000001137285001)

[3.13 Setting the iBMC Service Information](#_EN-US_TOPIC_0000001137285045)

[3.14 Querying the Power Consumption Information](#_EN-US_TOPIC_0000001137285087)

[3.15 Querying the System Information](#_EN-US_TOPIC_0000001137285131)

[3.16 Querying the Ethernet Port Information](#_EN-US_TOPIC_0000001137285175)

[3.17 Setting the Power Supply Status](#_EN-US_TOPIC_0000001137285003)

[3.18 Restarting the iBMC](#_EN-US_TOPIC_0000001137285047)

[3.19 Restarting the System](#_EN-US_TOPIC_0000001137285089)

[3.20 Querying the Asset Tag](#_EN-US_TOPIC_0000001137285133)

[3.21 Setting the Asset Tag](#_EN-US_TOPIC_0000001137285177)

[3.22 Querying the Virtual Media Information](#_EN-US_TOPIC_0000001137285005)

[3.23 Connecting to the Virtual Media](#_EN-US_TOPIC_0000001137285217)

[3.24 Disconnecting from the Virtual Media](#_EN-US_TOPIC_0000001137285091)

[3.25 Querying the Boot Device](#_EN-US_TOPIC_0000001137285135)

[3.26 Setting the Boot Device](#_EN-US_TOPIC_0000001137285179)

[3.27 Querying the Boot Sequence](#_EN-US_TOPIC_0000001137285007)

[3.28 Setting the Boot Sequence](#_EN-US_TOPIC_0000001137285049)

[3.29 Exporting the BIOS Configuration](#_EN-US_TOPIC_0000001137285093)

[3.30 Importing the BIOS Configuration](#_EN-US_TOPIC_0000001137285137)

[3.31 Restoring the Factory Settings](#_EN-US_TOPIC_0000001137285181)

[3.32 Restoring the Default BIOS Settings](#_EN-US_TOPIC_0000001137285009)

[3.33 Querying BIOS Settings](#_EN-US_TOPIC_0000001137285051)

[3.34 Modifying BIOS Settings](#_EN-US_TOPIC_0000001137285095)

[3.35 Querying the NTP Resource Information](#_EN-US_TOPIC_0000001137285139)

[3.36 Setting NTP Resource Attributes](#_EN-US_TOPIC_0000001137285183)

[3.37 Importing the NTP Group Key](#_EN-US_TOPIC_0000001137285011)

[3.38 Querying the SMTP Notification List](#_EN-US_TOPIC_0000001137285053)

[3.39 Setting the SMTP Notification Object](#_EN-US_TOPIC_0000001137285097)

[3.40 Querying the SMTP Resource Information](#_EN-US_TOPIC_0000001137285141)

[3.41 Setting SMTP Resource Attributes](#_EN-US_TOPIC_0000001137285185)

[3.42 Querying the SNMP Resource Information](#_EN-US_TOPIC_0000001137285013)

[3.43 Setting SNMP Resource Attributes](#_EN-US_TOPIC_0000001137285055)

[3.44 Querying the SNMP Trap Server](#_EN-US_TOPIC_0000001137285099)

[3.45 Setting the SNMP Trap Server](#_EN-US_TOPIC_0000001137285143)

[3.46 Querying the SNMP Trap Resource Information](#_EN-US_TOPIC_0000001137285187)

[3.47 Setting SNMP Trap Resource Attributes](#_EN-US_TOPIC_0000001137285015)

[3.48 Querying the Syslog Notification List](#_EN-US_TOPIC_0000001137285057)

[3.49 Setting the Syslog Notification Object](#_EN-US_TOPIC_0000001137285101)

[3.50 Querying the Syslog Resource Information](#_EN-US_TOPIC_0000001137285145)

[3.51 Setting the Syslog Resource Information](#_EN-US_TOPIC_0000001137285189)

[3.52 Querying the In-Band Firmware Version](#_EN-US_TOPIC_0000001137285017)

[3.53 Querying the Out-of-Band Firmware Version](#_EN-US_TOPIC_0000001137285059)

[3.54 Upgrading the In-Band Firmware](#_EN-US_TOPIC_0000001137285103)

[3.55 Upgrading the Out-of-Band Firmware](#_EN-US_TOPIC_0000001137285147)

[3.56 Querying the CPU Health Status](#_EN-US_TOPIC_0000001137285191)

[3.57 Querying the RAID Controller Card Health Status](#_EN-US_TOPIC_0000001137285019)

[3.58 Querying the Memory Health Status](#_EN-US_TOPIC_0000001137285061)

[3.59 Querying the Power Supply Health Status](#_EN-US_TOPIC_0000001137285105)

[3.60 Querying the Drive Health Status](#_EN-US_TOPIC_0000001137285149)

[3.61 Querying the NIC Health Status](#_EN-US_TOPIC_0000001137285193)

[3.62 Querying the Fan Health Status](#_EN-US_TOPIC_0000001137285021)

[3.63 Querying the CPU Information](#_EN-US_TOPIC_0000001137285063)

[3.64 Querying the RAID Controller Card Information](#_EN-US_TOPIC_0000001137285107)

[3.65 Querying the Memory Information](#_EN-US_TOPIC_0000001137285151)

[3.66 Querying the Power Supply Information](#_EN-US_TOPIC_0000001137285195)

[3.67 Querying the Drive Information](#_EN-US_TOPIC_0000001137285023)

[3.68 Querying the NIC Information](#_EN-US_TOPIC_0000001137285065)

[3.69 Querying the Fan Information](#_EN-US_TOPIC_0000001137285109)

[3.70 Querying the Configuration Result Resource of the Smart Provisioning Service](#_EN-US_TOPIC_0000001137285153)

[3.71 Modifying the Smart Provisioning Service Resource Attributes](#_EN-US_TOPIC_0000001137285197)

[3.72 Exporting the RAID Configuration of the Smart Provisioning Service](#_EN-US_TOPIC_0000001137285025)

[3.73 Creating the RAID Configuration of the Smart Provisioning Service](#_EN-US_TOPIC_0000001137285067)

[3.74 Deleting the RAID Configuration of the Smart Provisioning Service](#_EN-US_TOPIC_0000001137285111)

[3.75 Modifying the RAID Configuration of the Smart Provisioning Service](#_EN-US_TOPIC_0000001137285155)

[3.76 Querying the RAID Configuration Resource of the Smart Provisioning Service](#_EN-US_TOPIC_0000001137285199)

[3.77 Querying the OS Installation Configuration Resources of the Smart Provisioning Service](#_EN-US_TOPIC_0000001137285027)

[3.78 Creating the OS Installation Configuration for the Smart Provisioning Service](#_EN-US_TOPIC_0000001137285069)

[3.79 Restoring the Default Configuration of a Specified RAID Controller Card](#_EN-US_TOPIC_0000001137285113)

[3.80 Modifying the Information of a Specified RAID Controller Card Resource](#_EN-US_TOPIC_0000001137285157)

[3.81 Modifying the Attributes of a Specified Drive](#_EN-US_TOPIC_0000001137285201)

[3.82 Querying the Logical Drive Resource Information](#_EN-US_TOPIC_0000001137285029)

[3.83 Modifying the Logical Drive Resource Attributes](#_EN-US_TOPIC_0000001137285071)

[3.84 Initializing a Logical Drive](#_EN-US_TOPIC_0000001137285115)

[3.85 Deleting a Logical Drive](#_EN-US_TOPIC_0000001137285159)

[3.86 Creating a Logical Drive](#_EN-US_TOPIC_0000001137285203)

[3.87 Collecting iBMC Logs in One-click Mode](#_EN-US_TOPIC_0000001137285031)

[3.88 Downloading a BMC File](#_EN-US_TOPIC_0000001137285073)

[3.89 Uploading a File](#_EN-US_TOPIC_0000001137285117)

[3.90 Querying the License Service Information](#_EN-US_TOPIC_0000001137285161)

[3.91 Installing a License](#_EN-US_TOPIC_0000001137285205)

[3.92 Exporting a License](#_EN-US_TOPIC_0000001137285033)

[3.93 Deleting a License](#_EN-US_TOPIC_0000001137285075)

[3.94 Querying iBMC Network Port Information](#_EN-US_TOPIC_0000001137285119)

[3.95 Setting iBMC Network port Information](#_EN-US_TOPIC_0000001137285163)

[3.96 Querying LDAP Information](#_EN-US_TOPIC_0000001137285207)

[3.97 Modifying LDAP Information](#_EN-US_TOPIC_0000001137285035)

[3.98 Importing an LDAP certificate to the iBMC](#_EN-US_TOPIC_0000001137285077)

[3.99 Setting the LDAP Status](#_EN-US_TOPIC_0000001137285121)

[3.100 Querying Kerberos Service Resources](#_EN-US_TOPIC_0000001137285221)

[3.101 Changing the Kerberos Enabling Status](#_EN-US_TOPIC_0000001137285209)

[3.102 Querying the Kerberos Domain Controller Information](#_EN-US_TOPIC_0000001137285219)

[3.103 Modifying the Kerberos Domain Controller Information](#_EN-US_TOPIC_0000001137285079)

[3.104 Importing a Kerberos Domain Controller Key Table](#_EN-US_TOPIC_0000001137285123)

[3.105 Querying a Diagnostic Service Resource](#_EN-US_TOPIC_0000001137285165)

[3.106 Modifying Diagnostic Service Resource Information](#_EN-US_TOPIC_0000001137285211)

[3.107 General Interface](#_EN-US_TOPIC_0000001137285037)

[3.108 Importing the Root Certificate of a Remote HTTPS Server](#_EN-US_TOPIC_0000001137285229)

[3.109 Deleting the Root Certificate of a Remote HTTPS Server](#_EN-US_TOPIC_0000001137285125)

[3.110 Importing a CRL of a Remote HTTPS Server](#_EN-US_TOPIC_0000001137285167)

[3.111 Querying the Certificate Verification Switch Status of the Remote HTTPS Server](#_EN-US_TOPIC_0000001137285213)

[3.112 Enabling/Disabling the Certificate Verification of the Remote HTTPS Server](#_EN-US_TOPIC_0000001137285039)

## Viewing the Help Information

Function

View the command help information.

Format

**Get-Help** *<Name>* **-Full**

Parameters

| Parameter | Description |
| --- | --- |
| *<Name>* | Specifies the parameter name. This parameter is mandatory. |
| -Full | Specifies the detailed description of the command. This parameter is optional. |

Usage Guidelines

None

Example

View the help information of the **Connect-iBMC** command.

PS C:\> **Get-Help Connect-iBMC -Full**

The command output of **Connect-iBMC** is as follows:

NAME   
 Connect-iBMC   
SYNOPSIS   
 Connect to iBMC Servers and initialize sessions used by other cmdlets.   
SYNTAX   
 Connect-iBMC [-Address] <String[]> [-Username] <String[]> [-Password] <String[]> [-TrustCert] [<CommonParameters>]   
 Connect-iBMC [-Address] <String[]> [-Credential] <PSCredential[]> [-TrustCert] [<CommonParameters>]   
DESCRIPTION   
 Initialize sessions for one or multiple iBMC servers and. This cmdlet has following parameters:   
 - Address - Holds the iBMC server IP/hostname.   
 - Username - Holds the iBMC server username.   
 - Password - Holds the iBMC server password.   
 - Credential - Holds the iBMC server Credential.   
 - TrustCert - Using this bypasses the server certificate authentication.   
PARAMETERS   
 -Address <String[]>   
 IP address or Hostname of the iBMC server.   
 Required? true   
 Position? 1   
 Default value   
 Accept pipeline input? true (ByValue, ByPropertyName)   
 Accept wildcard characters? false   
 -Username <String[]>   
 Username of iBMC account to access the iBMC server.   
 Required? true   
 Position? 2   
 Default value   
 Accept pipeline input? true (ByPropertyName)   
 Accept wildcard characters? false   
 -Password <String[]>   
 Password of iBMC account to access the iBMC server.   
 Required? true   
 Position? 3   
 Default value   
 Accept pipeline input? true (ByPropertyName)   
 Accept wildcard characters? false   
 -Credential <PSCredential[]>   
 PowerShell PSCredential object having username and passwword of iBMC account to access the iBMC.   
 Required? true   
 Position? 2   
 Default value   
 Accept pipeline input? true (ByPropertyName)   
 Accept wildcard characters? false   
 -TrustCert [<SwitchParameter>]   
 If this switch parameter is present then server certificate authentication is disabled for this iBMC session.   
 If not present, server certificate is enabled by default.   
 Required? false   
 Position? named   
 Default value False   
 Accept pipeline input? false   
 Accept wildcard characters? false   
 <CommonParameters>   
 This cmdlet supports the common parameters: Verbose, Debug,   
 ErrorAction, ErrorVariable, WarningAction, WarningVariable,   
 OutBuffer, PipelineVariable, and OutVariable. For more information, see   
 about\_CommonParameters (https:/go.microsoft.com/fwlink/?LinkID=113216).   
INPUTS   
OUTPUTS   
 RedfishSession[]   
 Returns the created RedfishSession if cmdlet executes successfully.   
 In case of an error or warning, exception will be returned.   
 -------------------------- EXAMPLE 1 --------------------------   
 PS C:\>$sessions = Connect-iBMC -Address 10.1.1.2 -Username root -Password password   
 PS C:\> $sessions   
 -------------------------- EXAMPLE 2 --------------------------   
 PS C:\>$credential = Get-Credential   
 PS C:\> $sessions = Connect-iBMC -Address 10.1.1.2 -Credential $credential   
 PS C:\> $sessions   
 -------------------------- EXAMPLE 3 --------------------------   
 PS C:\>$sessions = Connect-iBMC -Address "10.1.1.2,5,8" -Username root -Password password   
 PS C:\> $sessions   
 -------------------------- EXAMPLE 4 --------------------------   
 PS C:\>$sessions = Connect-iBMC -Address 10.1.1.2-10 -Username root -Password password   
 PS C:\> $sessions   
 -------------------------- EXAMPLE 5 --------------------------   
 PS C:\>$sessions = Connect-iBMC -Address 10.1.1.2,10.1.1.3 -Username root -Password password   
 PS C:\> $sessions   
 -------------------------- EXAMPLE 6 --------------------------   
 PS C:\>$sessions = Connect-iBMC -Address 10.1.1.2,10.1.1.3 -Username user1,user2 -Password password1,password2   
 PS C:\> $sessions   
 -------------------------- EXAMPLE 7 --------------------------   
 PS C:\>$sessions = Connect-iBMC -Address \*\*\*\*::\*\*\*\* -Username root -Password password   
 PS C:\> $sessions   
 This example shows how to connect to a bmc server using ipv6   
 -------------------------- EXAMPLE 8 --------------------------   
 PS C:\>$sessions = Connect-iBMC -Address "[\*\*\*\*::\*\*\*\*]:8080" -Username root -Password password   
 PS C:\> $sessions   
 This example shows how to connect to a bmc server using ipv6 and port   
 -------------------------- EXAMPLE 9 --------------------------   
 PS C:\>$sessions = Connect-iBMC -Address "\*\*\*\*::\*\*\*\*,201A" -Username root -Password password   
 PS C:\> $sessions   
 This example shows how to connect to multiple bmc server using "," seperated ipv6 addresses   
 -------------------------- EXAMPLE 10 --------------------------   
 PS C:\>$sessions = Connect-iBMC -Address "\*\*\*\*::\*\*\*\*-201A" -Username root -Password password   
 PS C:\> $sessions   
 This example shows how to connect to multiple bmc server using "-" seperated ipv6 addresses   
RELATED LINKS

## Querying the xFusion-iBMC-Cmdlets Version Information

Function

Query the xFusion-iBMC-Cmdlets version information.

Format

**Get-iBMCModuleVersion**

Parameters

None

Usage Guidelines

None

Example

Query the current xFusion-iBMC-Cmdlets version.

PS C:\> **Get-iBMCModuleVersion**   
   
Name Version Path   
---- ------- ----   
xFusion-iBMC-Cmdlets 1.3.6 C:\WINDOWS\system32\WindowsPowerShell\v1.0\Modules\xFusion-iBMC-Cmdlets\xFusion-iBMC-Cmdle...

## Connecting to the iBMC

Function

Connect to the iBMC.

Format

**Connect-iBMC -Address** <*Address*> **-Username** <*Username*> **-Password** <*Password*> **-TrustCert**

**Connect-iBMC -Address** <*Address*> **-Credential** <*$Credential*> **-TrustCert**

Parameters

| Parameter | Description |
| --- | --- |
| <*Address*> | Specifies the IPv4/IPv6 address or domain name of the server iBMC. This parameter is mandatory.  Multiple IP addresses can be separated by commas (,) or hyphens (-). If an IPv6 address is used, the IPv6 address must be enclosed in parentheses, for example, **[xxxx::xxxx]**. |
| <*Username*> | Specifies the iBMC administrator user name. This parameter is optional.  Multiple user names can be separated by commas (,). |
| *<Password>* | Specifies the iBMC administrator password. This parameter is optional.  Multiple passwords can be separated by commas (,). |
| <$*Credential*> | Specifies the encrypted connection to the iBMC. This parameter is optional.  For details about how to create an iBMC server credential, see [Example](#section1464932193810). |
| -TrustCert | Specifies bypassing server certificate authentication. This parameter is optional. |
| NOTE   * When the iBMC is connected in plain text mode, the **Username** and **Password** parameters are mandatory. * When the iBMC is connected in encryption mode, the **Credential** parameter is mandatory. | |

Usage Guidelines

None

Example

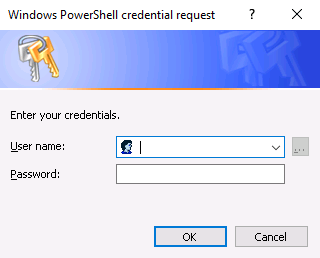
# (Recommended) Connect to the iBMC in encryption mode and create a session.

1. Create an iBMC server credential.

PS C:\>**$credential = Get-Credential**   
 cmdlet Get-Credential at command pipeline position 1   
 Supply values for the following parameters:   
 Credential

The **Windows PowerShell credential request** dialog box is displayed, as shown in Figure 3-1.

Windows PowerShell credential request



1. In the **Windows PowerShell credential request** dialog box, enter the iBMC user name and password.
2. Click **OK**.
3. Connect to the iBMC in encryption mode and create a session.

PS C:\>**$session = Connect-iBMC -Address 192.168.1.1 -Credential $Credential -TrustCert**

1. View iBMC information in the session.

PS C:\>**$session**   
Id : 1   
 Name : Manager   
 ManagerType : BMC   
 FirmwareVersion : 3.04   
 UUID : 3EBAF6CC-69CD-11E7-BF57-68CC6E3CDB29   
 Model : iBMC   
 Health : OK   
 State : Enabled   
 DateTime : 2018-12-16T15:17:52+00:00   
 DateTimeLocalOffset : GMT   
 Address : 10.1.1.2   
 BaseUri : https://10.1.1.2   
 Location : /redfish/v1/SessionService/Sessions/7ca5ad94c4dfc85e   
 Alive : True   
 AuthToken : 8b5f6599cdc89a14d36f5299a14952d9   
 TrustCert : True

# Connect to a single iBMC and create a session.

1. Create a session and connect to the iBMC.

PS C:\>**$session = Connect-iBMC -Address 10.1.1.2 -Username username -Password password -TrustCert**

1. View iBMC information in the session.

PS C:\>**$session**   
 Id : 1   
 Name : Manager   
 ManagerType : BMC   
 FirmwareVersion : 3.04   
 UUID : 3EBAF6CC-69CD-11E7-BF57-68CC6E3CDB29   
 Model : iBMC   
 Health : OK   
 State : Enabled   
 DateTime : 2018-12-16T15:17:52+00:00   
 DateTimeLocalOffset : GMT   
 Address : 10.1.1.2   
 BaseUri : https://10.1.1.2   
 Location : /redfish/v1/SessionService/Sessions/7ca5ad94c4dfc85e   
 Alive : True   
 AuthToken : 8b5f6599cdc89a14d36f5299a14952d9   
 TrustCert : True

# Connect to multiple iBMCs and create a session.

1. Create a session and connect to multiple iBMCs.

PS C:\> $sessions = **Connect-iBMC -Address 10.1.1.2,10.1.1.3 -Username root -Password password**

1. View iBMC information in the session.

PS C:\> **$sessions**   
   
 Id : 1   
 Name : Manager   
 ManagerType : BMC   
 FirmwareVersion : 3.04   
 UUID : 3EBAF6CC-69CD-11E7-BF57-68CC6E3CDB29   
 Model : iBMC   
 Health : OK   
 State : Enabled   
 DateTime : 2018-12-16T15:17:52+00:00   
 DateTimeLocalOffset : GMT   
 Address : 10.1.1.2   
 BaseUri : https://10.1.1.2   
 Location : /redfish/v1/SessionService/Sessions/7ca5ad94c4dfc85e   
 Alive : True   
 AuthToken : 8b5f6599cdc89a14d36f5299a14952d9   
 TrustCert : True   
   
 Id : 2   
 Name : Manager   
 ManagerType : BMC   
 FirmwareVersion : 3.04   
 UUID : 3EBAF6CC-69CD-11E7-BF57-68CC6E3CDB29   
 Model : iBMC   
 Health : OK   
 State : Enabled   
 DateTime : 2018-12-16T15:17:52+00:00   
 DateTimeLocalOffset : GMT   
 Address : 10.1.1.3   
 BaseUri : https://10.1.1.3   
 Location : /redfish/v1/SessionService/Sessions/7ca5ad94c4dfc85e   
 Alive : True   
 AuthToken : 8b5f6599cdc89a14d36f5299a14952d9   
 TrustCert : True

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Id | Specifies the session ID, which uniquely identifies the session resource. |
| Name | Specifies the session name |
| ManagerType | Specifies the session resource type. |
| FirmwareVersion | Specifies the session resource firmware version. |
| UUID | Specifies the universally unique identifier of the session resource. |
| Model | Specifies the session resource model. |
| Health | Specifies the session resource health status. |
| State | Specifies the session resource enabling status. |
| DateTime | Specifies the session resource system time. |
| DateTimeLocalOffset | Specifies the session resource time zone. |
| Address | Specifies the server IP address. |
| BaseUri | Specifies the network address. |
| Location | Specifies the access address. |
| Alive | Specifies the connection status. |
| AuthToken | Specifies the authentication key. |
| TrustCert | Specifies whether to bypass server certificate authentication. |

## Disconnecting from the iBMC

Function

Disconnect from the iBMC.

Format

**Disconnect-iBMC -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Disconnect from the iBMC.

PS C:\>**Disconnect-iBMC -Session $session**   
Id : 1   
Name : Manager   
ManagerType : BMC   
FirmwareVersion : 3.16   
UUID : 02AB0D57-4857-9D57-E811-92DD90161F12   
Model : iBMC   
Health : OK   
State : Enabled   
DateTime : 2018-12-29T02:09:19+00:00   
DateTimeLocalOffset : GMT   
Address : 10.1.1.2   
BaseUri : https://10.1.1.2   
Location : /redfish/v1/SessionService/Sessions/4c8c5f281cb30981   
Alive : False   
AuthToken : ea9a4c9dfd25706f52343a486afe203f   
TrustCert : True

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Id | Specifies the session ID, which uniquely identifies the session resource. |
| Name | Specifies the session name |
| ManagerType | Specifies the session resource type. |
| FirmwareVersion | Specifies the session resource firmware version. |
| UUID | Specifies the universally unique identifier of the session resource. |
| Model | Specifies the session resource model. |
| Health | Specifies the session resource health status. |
| State | Specifies the session resource enabling status. |
| DateTime | Specifies the session resource system time. |
| DateTimeLocalOffset | Specifies the session resource time zone. |
| Address | Specifies the server IP address. |
| BaseUri | Specifies the network address. |
| Location | Specifies the access address. |
| Alive | Specifies the connection status. |
| AuthToken | Specifies the authentication key. |
| TrustCert | Specifies whether to bypass server certificate authentication. |

## Testing the iBMC Connection Status

Function

Test the iBMC connection status.

Format

**Test-iBMCConnect -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Test the iBMC connection status.

PS C:\> **Test-iBMCConnect -Session $session**   
   
Id : 1   
Name : Manager   
ManagerType : BMC   
FirmwareVersion : 3.04   
UUID : 3EBAF6CC-69CD-11E7-BF57-68CC6E3CDB29   
Model : iBMC   
Health : OK   
State : Enabled   
DateTime : 2018-12-20T17:34:28+00:00   
DateTimeLocalOffset : GMT   
Address : 10.1.1.2   
BaseUri : https://10.1.1.2   
Location : /redfish/v1/SessionService/Sessions/19b0f46c5e524f1f   
Alive : True   
AuthToken : f239b99e1a15b0f6f32213c2fe0e2eab   
TrustCert : True

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Id | Specifies the session ID, which uniquely identifies the session resource. |
| Name | Specifies the session name |
| ManagerType | Specifies the session resource type. |
| FirmwareVersion | Specifies the session resource firmware version. |
| UUID | Specifies the universally unique identifier of the session resource. |
| Model | Specifies the session resource model. |
| Health | Specifies the session resource health status. |
| State | Specifies the session resource enabling status. |
| DateTime | Specifies the session resource system time. |
| DateTimeLocalOffset | Specifies the session resource time zone. |
| Address | Specifies the server IP address. |
| BaseUri | Specifies the network address. |
| Location | Specifies the access address. |
| Alive | Specifies the connection status. |
| AuthToken | Specifies the authentication key. |
| TrustCert | Specifies whether to bypass server certificate authentication. |

## Querying the Session Service Information

Function

Query the timeout interval of the current session.

Format

**Get-iBMCSessionTimeout** **-Session** *<$session>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the session timeout interval.

PS C:\> **$SessionTimeout = Get-iBMCSessionTimeout -Session $session**   
PS C:\> **$SessionTimeout**   
   
Host : 10.1.1.2   
SessionTimeout : 600

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Host |
| SessionTimeout | Specifies the session timeout interval, in seconds. |

## Modifying the Session Service Information

Function

Modify the timeout interval of the current session service on the server.

Format

**Set-iBMCSessionTimeout** **-Session** *<$session>* **-Timeout** *<Timeout>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<Timeout>* | Specifies the session timeout interval. This parameter is mandatory. | The value is an integer ranging from 30 to 86400, and the unit is second. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Change the session timeout interval to 600 seconds.

PS C:\> **Set-iBMCSessionTimeout -Session $session -Timeout 600**

## Querying the User List

Function

Query the iBMC user list.

Format

**Get-iBMCUser -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the command session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the iBMC user list.

PS C:\> **$tasks = Get-iBMCUser -Session $session**   
PS C:\> **$tasks**   
   
Host : 10.1.1.2   
Id : 2   
Name : User Account   
UserName : root   
RoleId : Administrator   
Locked : False   
Enabled : True   
Oem : @{xFusion=}   
   
Host : 10.1.1.2   
Id : 3   
Name : User Account   
UserName : 12312ada   
RoleId : Noaccess   
Locked : False   
Enabled : False   
Oem : @{xFusion=}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the user ID. |
| Name | Specifies the account name. |
| UserName | Specifies the user name. |
| RoleId | Specifies the ID of the role resource configured for the account. |
| Locked | Specifies that the account service is automatically locked because the lock threshold is exceeded. When this item is set to **True**, the account is locked. The user administrator can set this item to **False** to unlock the account. |
| Enabled | Specifies whether to disable the account that has the user information deletion function. When this item is set to **True**, the user can log in. When this item is set to **False**, the account is disabled and the user cannot log in. |
| Oem | Specifies the customized attribute. |

## Creating a User

Function

Create a user.

Format

**Add-iBMCUser -Session** *<$session>* **-Username <***Username****>* -Password** *<Password>* **-Role** *<Role>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| **<***Username****>*** | Specifies the name of the newly created user. This parameter is mandatory.  Multiple user names can be separated by commas (,). | - |
| *<Password>* | Specifies the password of the newly created user. This parameter is mandatory.  Multiple passwords can be separated by commas (,). | - |
| *<Role>* | Specifies the role of the newly created user. This parameter is mandatory.  Multiple roles can be separated by commas (,). | * Administrator * Operator * Commonuser * Noaccess * CustomRole1 * CustomRole2 * CustomRole3 * CustomRole4 |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Set the password of the new user to **password**.

PS C:\> **$pwd = ConvertTo-SecureString -String password -AsPlainText -Force**

1. Create a user.

PS C:\> **,$sessions | Add-iBMCUser -Username new-user,new-user2 -Password $pwd,$pwd -Role Operator,Administrator**   
   
Host : 10.1.1.2   
Id : 4   
Name : User Account   
UserName : new-user   
RoleId : Operator   
Locked : False   
Enabled : True   
Oem : @{xFusion=}   
   
Host : 10.1.1.2   
Id : 5   
Name : User Account   
UserName : new-user2   
RoleId : Administrator   
Locked : False   
Enabled : True   
Oem : @{xFusion=}

## Modifying a User

Function

Modify a user.

Format

**Set-iBMCUser -Session** *<$session>* **-Username*<****Username****>*-NewUsername** *<NewUsername>* **-NewPassword** *<NewPassword>* **-NewRole** *<Role>* **-Enabled** *<Enabled>* **-Unlocked***<Unlocked >*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| **<***Username***>** | Specifies the name of the user to be modified. This parameter is mandatory. | - |
| *<NewUsername>* | Specifies the new name of the user to be modified. This parameter is optional. | - |
| *<NewPassword>* | Specifies the new password of the user to be modified. This parameter is optional. | - |
| *<Role>* | Specifies the new role of the user to be modified. This parameter is optional. | * Administrator * Operator * Commonuser * Noaccess * CustomRole1 * CustomRole2 * CustomRole3 * CustomRole4 |
| *<Enabled>* | Specifies the enabling status of the user. This parameter is optional. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<Unlocked >* | Specifies the locking status of the user. This parameter is optional. | * **$true** or **1**: unlocked * **$false** or **0**: locked |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Set a new password.

PS C:\> **$newPwd = ConvertTo-SecureString -String *new-password* -AsPlainText -Force**

1. Change the user name and password.

PS C:\> **,$sessions | Set-iBMCUser -Username username -NewUsername new-user2 -NewPassword $newPwd -NewRole Administrator**   
   
Host : 10.1.1.2   
Id : 12   
Name : User Account   
UserName : powershell   
RoleId : Operator   
Locked : True   
Enabled : True   
Oem : @{xFusion=}

## Removing a User

Function

Remove a user.

Format

**Remove-iBMCUser -Session** *<$session>* **-Username <***Username***>**

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |
| **<***Username***>** | Specifies the name of the user to be removed. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Remove the user whose name is **user**.

PS C:\> **,$session | Remove-iBMCUser -Username user**

## Querying the iBMC Service Information

Function

Query the iBMC service information.

Format

**Get-iBMCServices -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the iBMC service information.

PS C:\> **$Services = Get-iBMCServices -Session $session**   
PS C:\> **$Services**   
Host : 10.1.1.2   
HTTP : @{ProtocolEnabled=True; Port=80}   
HTTPS : @{ProtocolEnabled=True; Port=443}   
SNMP : @{ProtocolEnabled=True; Port=161}   
VirtualMedia : @{ProtocolEnabled=True; Port=8208}   
IPMI : @{ProtocolEnabled=True; Port=623}   
SSH : @{ProtocolEnabled=True; Port=22}   
KVMIP : @{ProtocolEnabled=True; Port=2198}   
SSDP : @{ProtocolEnabled=False; Port=1900; NotifyMulticastIntervalSeconds=600; NotifyTTL=2; NotifyIPv6Scope=Site}   
VNC : @{ProtocolEnabled=False; Port=5900}   
Video : @{ProtocolEnabled=True; Port=2199}   
NAT :

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| HTTP/HTTPS/SNMP/VirtualMedia/IPMI/SSH/KVMIP/SSDP/VNC/Video/NAT | Specifies the service information supported by the iBMC. Services that are not supported by the switch module are displayed as null. |
| ProtocolEnabled | Specifies the service enabling status. |
| Port | Specifies the service port number. |
| NotifyMulticastIntervalSeconds | Specifies the multicast interval of the SSDP service message. |
| NotifyTTL | Specifies the validity period of the SSDP service message. |
| NotifyIPv6Scope | Specifies the IPv6 multicast range of the SSDP service message.   * **Link**: local link range * **Site**: local site range * **Organization**: local organization range |

## Setting the iBMC Service Information

Function

Set the iBMC service information.

Format

**Set-iBMCService -Session** *<$Session>* **-ServiceName** *<ServiceName>* **-Enabled** <*Enabled>* **-Port** *<Port>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *$session* | Specifies the session. This parameter is mandatory. | - |
| *<ServiceName>* | Specifies the service information supported by the iBMC. This parameter is mandatory. | * HTTP * HTTPS * SNMP * VirtualMedia * IPMI * SSH * KVMIP * VNC * Video * NAT |
| *<Enabled>* | Specifies the service enabling status. This parameter is mandatory. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<Port>* | Specifies the service port number. This parameter is mandatory. | 1 to 65535 |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Set the iBMC service information.

PS C:\>**Set-iBMCService -Session $session -ServiceName VNC -Enabled $true -Port 5900**

## Querying the Power Consumption Information

Function

Query the power consumption information.

Format

**Get-iBMCPowerInfo -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the power consumption information.

PS C:\> **$tasks = Get-iBMCPowerInfo -Session $sessions**   
PS C:\> **$tasks**   
   
Host : 10.1.1.2   
Id : 0   
Name : System Power Control 1   
PowerConsumedWatts : 222 Watts   
MaxConsumedWatts : 432 Watts   
MinConsumedWatts : 18 Watts   
AverageConsumedWatts : 183 Watts

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the power supply ID. |
| Name | Specifies the power supply name. |
| PowerConsumedWatts | Specifies the current power. |
| MaxConsumedWatts | Specifies the maximum power consumption. |
| MinConsumedWatts | Specifies the minimum power consumption. |
| AverageConsumedWatts | Specifies the average power consumption. |

## Querying the System Information

Function

Query the system information.

Format

**Get-iBMCSystemInfo -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the system information.

PS C:\> **$System = Get-iBMCSystemInfo -Session $session**   
PS C:\> **$System**   
   
Host : 10.1.1.2   
Id : 1   
Name : Computer System   
AssetTag : my test   
Manufacturer : xFusion   
Model : 2288H V5   
SerialNumber : 2102311TYBN0J3000293   
UUID : 877AA970-58F9-8432-E811-80345C184638   
HostName :   
PartNumber : 02311TYB   
HostingRole : {ApplicationServer}   
Status : @{State=Disabled; Health=OK}   
PowerState : Off   
Boot : @{BootSourceOverrideTarget=Pxe; BootSourceOverrideEnabled=Continuous; BootSourceOverrideMode=Legacy; BootSourceOverride   
 Target@Redfish.AllowableValues=System.Object[]}   
TrustedModules :   
BiosVersion : 0.81   
ProcessorSummary : @{Count=2; Model=Central Processor; Status=}   
MemorySummary : @{TotalSystemMemoryGiB=128; Status=}   
PCIeDevices : {}   
PCIeFunctions : {}   
Oem : @{xFusion=}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the system ID. |
| Name | Specifies the system name. |
| AssetTag | Specifies the system asset tag. |
| Manufacturer | Specifies the system manufacturer. |
| Model | Specifies the system model. |
| SerialNumber | Specifies the system serial number. |
| UUID | Specifies the globally unique ID of the system. |
| HostName | Specifies the host name of the system. |
| PartNumber | Specifies the part number of the memory. |
| HostingRole | Specifies the host role of the system. |
| Status | Specifies the system status. The options are as follows:   * **State**: system resource enabling status * **Health**: system resource health status |
| PowerState | Specifies the system power status. |
| Boot | Specifies the boot-related information. |
| BootSourceOverrideTarget | Specifies the current boot device of the system. |
| BootSourceOverrideEnabled | Specifies the enabling status of the system boot parameter. The options are as follows:   * **Disabled**: The system boot parameter does not take effect. * **Once**: The system boot parameter takes effect only once, that is, it takes effect upon the next restart. * **Continuous**: The system boot parameter takes effect permanently. |
| BootSourceOverrideMode | Specifies the system boot mode. The options are as follows:   * Legacy * UEFI |
| TrustedModules | Specifies the trusted module of system resources. |
| BiosVersion | Specifies the BIOS version. |
| ProcessorSummary | Specifies the CPU information. |
| Count | Specifies the number of CPUs. |
| Model | Specifies the CPU model. |
| Status | Specifies the CPU health status. |
| MemorySummary | Specifies the memory information. |
| TotalSystemMemoryGiB | Specifies the total system memory capacity. The unit is GB. |
| Status | Specifies the memory health status. |
| PCIeDevices | Specifies the access path of the PCIe device resource node. |
| PCIeFunctions | Specifies the access path of the PCIe function resource node. |
| Oem | Specifies the customized attribute. |

## Querying the Ethernet Port Information

Function

Query the Ethernet port information.

Format

**Get-iBMCSystemNetworkSettings -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

You need to install the iBMA of the corresponding version on the OS to obtain complete Ethernet port information.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the Ethernet port information.

PS C:\> **$Interfaces = Get-iBMCSystemNetworkSettings -Session $session**   
PS C:\> **$Interfaces**   
   
Host : 10.1.1.2   
Id : mainboardLOMPort1   
Name : System Ethernet Interface   
PermanentMACAddress : \*\*:\*\*:\*\*:\*\*:\*\*:\*\*   
MACAddress : \*\*:\*\*:\*\*:\*\*:\*\*:\*\*   
LinkStatus : LinkUp   
IPv4Addresses : {@{Address=10.1.1.20; SubnetMask=255.255.0.0; Gateway=10.1.0.1; AddressOrigin=}}   
IPv6Addresses : {@{Address=\*\*\*\*::\*\*\*; PrefixLength=64; AddressOrigin=SLAAC; AddressState=},   
 @{Address=\*\*\*\*::\*\*\*;PrefixLength=64; AddressOrigin=SLAAC; AddressState=},   
 @{Address=fe80::4a57:2ff:feab:d5a; PrefixLength=64; AddressOrigin=Static; AddressState=}}   
IPv6DefaultGateway : fe80::525d:acff:feed:5c27   
InterfaceType : Physical   
BandwidthUsage : 0   
BDF : 0000:1a:00.0

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the network port ID. |
| Name | Specifies the network port name. |
| PermanentMACAddress | Specifies the permanent MAC address of the network port. |
| MACAddress | Specifies the MAC address. |
| LinkStatus | Specifies the network port link status. |
| IPv4Addresses | Specifies the IPv4 address. |
| IPv6Addresses | Specifies the IPv6 address. |
| IPv6DefaultGateway | Specifies the default IPv6 gateway. |
| InterfaceType | Specifies the network port type.   * Physical * Virtual |
| BandwidthUsage | Specifies the bandwidth usage. |
| BDF | Specifies the BDF number. |

## Setting the Power Supply Status

Function

Control the power-on and power-off status of the FRU.



This operation may affect the proper running of the OS.

Format

**Set-iBMCFruControl -Session** *<$session>* ***-FRU*** *<-FRU>* **-ControlType** *<ControlType>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<-FRU*> | Specifies the FRU value. This parameter is mandatory. | * OS * Base * Fabric * FC |
| *<ControlType>* | Specifies the power control type. This parameter is mandatory. | * **On**: power-on * **GracefulShutdown**: graceful power-off * **ForceRestart**: forcible restart * **Nmi**: non-maskable interrupt * **ForcePowerCycle**: forcible power-off and power-on |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Set the server to be powered off gracefully.

PS C:\> **Set-iBMCFruControl -Session $session -FRU OS -ControlType GracefulShutdown**

## Restarting the iBMC

Function

Restart the iBMC.

Format

**Reset-iBMC -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Restart the iBMC.

PS C:\> **Reset-iBMC -Session $session**

## Restarting the System

Function

Control the power-on and power-off status of the server.



This operation may affect the proper running of the OS.

Format

**Set-iBMCServerPower -Session** *<$session>* **-ResetType** *<ResetType>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<ResetType>* | Specifies the power control type. This parameter is mandatory. | * **On**: power-on * **ForceOff**: forcible power-off * **GracefulShutdown**: graceful power-off * **ForceRestart**: forcible restart * **Nmi**: non-maskable interrupt * **ForcePowerCycle**: forcible power-off and power-on |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Restart the system.

PS C:\> **Set-iBMCServerPower -Session $session -ResetType ForceRestart**

## Querying the Asset Tag

Function

Query the asset tag.

Format

**Get-iBMCAssetTag -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the asset tag.

PS C:\> **$AssetTag = Get-iBMCAssetTag -Session $session**   
PS C:\> **$AssetTag | fl**   
   
Host : 10.1.1.2   
AssetTag : powershell-asset-tag

## Setting the Asset Tag

Function

Set the asset tag.

Format

**Set-iBMCAssetTag -Session** *<$session>* **-AssetTag** *<AssetTag>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<AssetTag>* | Specifies the asset tag. This parameter is mandatory. | Value range: The value is a string of 1 to 48 characters, consisting of digits, letters, and special characters.  If you want to set this parameter to null, set this parameter to a null character string. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Set the asset tag to **powershell-asset-tag**.

PS C:\> **Set-iBMCAssetTag -Session $session -AssetTag 'powershell-asset-tag'**

## Querying the Virtual Media Information

Function

Query the virtual media information.

Format

**Get-iBMCVirtualMedia -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the virtual media information.

PS C:\> $VirtualMedia = **Get-iBMCVirtualMedia -Session $session**   
PS C:\> $VirtualMedia   
   
Host : 10.1.1.2   
Id : CD   
Name : VirtualMedia   
MediaTypes : {}   
Image :   
ImageName :   
ConnectedVia : NotConnected   
Inserted : False

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the virtual media resource ID. |
| Name | Specifies the virtual media resource name. |
| MediaTypes | Specifies the media image type.   * CD * Floppy * USBStick * DVD |
| Image | Specifies the image URI. |
| ImageName | Specifies the image name. |
| ConnectedVia | Specifies the connection mode.   * NotConnected * URI * Applet * Oem |
| Inserted | Specifies whether to insert the virtual media. |

## Connecting to the Virtual Media

Function

Connect to the virtual media.

Format

**Connect-iBMCVirtualMedia -Session** *<$session>* **-ImageFilePath** *<ImageFilePath>* **-SecureEnabled**

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<ImageFilePath>* | Specifies the virtual media image URL. This parameter is mandatory. | Currently, only URLs that use the NFS, CIFS, and HTTPS protocols are supported. |
| -SecureEnabled | (Optional) Whether to enable the interactive function. The function is enabled when this parameter is added to the command. You are advised to enable this function when using a remote directory to ensure operation security. | - |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Connect to the virtual media of the NFS protocol.

PS C:\> **Connect-iBMCVirtualMedia -Session $session-ImageFilePath 'cifs://user:password@10.1.1.2/C$/iso/SmartProvisioning-V112.iso'**   
   
Host : 10.1.1.2   
Id : 1   
Name : vmm connect task   
ActivityName : [10.1.1.3] vmm connect task   
TaskState : Completed   
StartTime : 2018-12-29T02:36:21+00:00   
EndTime : 2018-12-29T02:36:34+00:00   
TaskStatus : OK   
TaskPercent :

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the task ID. |
| Name | Specifies the task name. |
| ActivityName | Specifies the virtual media connection status. |
| TaskState | Specifies the task execution progress. |
| StartTime | Specifies the task start time. |
| EndTime | Specifies the task end time. |
| TaskStatus | Specifies the task execution status. |
| TaskPercent | Specifies the task completion rate. |

## Disconnecting from the Virtual Media

Function

Disconnect from the virtual media.

Format

**Disconnect-iBMCVirtualMedia -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Disconnect from the virtual media.

PS C:\> **$Tasks = Disconnect-iBMCVirtualMedia -Session $session**   
PS C:\> **$Tasks**   
   
Host : 10.1.1.2   
Id : 4   
Name : vmm disconnect status task   
ActivityName : [10.1.1.2] vmm disconnect status task   
TaskState : Completed   
StartTime : 2018-11-14T18:05:20+08:00   
EndTime : 2018-11-14T18:05:20+08:00   
TaskStatus : OK   
TaskPercent :

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the task ID. |
| Name | Specifies the task name. |
| ActivityName | Specifies the virtual media connection status. |
| TaskState | Specifies the task execution progress. |
| StartTime | Specifies the task start time. |
| EndTime | Specifies the task end time. |
| TaskStatus | Specifies the task execution status. |
| TaskPercent | Specifies the task completion rate. |

## Querying the Boot Device

Function

Query the boot device.

Format

**Get-iBMCBootSourceOverride -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the boot device.

PS C:\> **$BootSourceOverride = Get-iBMCBootSourceOverride -Session $session**   
PS C:\> **$BootSourceOverride**   
   
Host BootSourceOverrideTarget BootSourceOverrideEnabled   
---- ------------------------ -------------------------   
10.1.1.2 None Disabled

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| BootSourceOverrideTarget | Specifies the current boot device of the system. The options are as follows:   * None * Pxe * Floppy * Cd * Hdd * BiosSetup |
| BootSourceOverrideEnabled | Specifies the enabling status of the system boot parameter. The options are as follows:   * **Disabled**: The system boot parameter does not take effect. * **Once**: The system boot parameter takes effect only once, that is, it takes effect upon the next restart. * **Continuous**: The system boot parameter takes effect permanently. |

## Setting the Boot Device

Function

Set the boot device.

Format

**Set-iBMCBootSourceOverride -Session** *<$session>* **-BootSourceOverrideTarget** *<BootSourceOverrideTarget>* **-BootSourceOverrideEnabled** *<BootSourceOverrideEnabled>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<BootSourceOverrideTarget>* | Specifies the current boot device of the system. This parameter is mandatory. | * None * Pxe * Floppy * Cd * Hdd * BiosSetup |
| *<BootSourceOverrideEnabled*> | Specifies the enabling status of the system boot parameter. This parameter is mandatory. | * **Disabled**: The system boot parameter does not take effect. * **Once**: The system boot parameter takes effect only once, that is, it takes effect upon the next restart. * **Continuous**: The system boot parameter takes effect permanently. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Set the boot device.

PS C:\> **Set-iBMCBootSourceOverride -Session $session -BootSourceOverrideTarget @('Pxe','Hdd') -BootSourceOverrideEnabled @('Once', 'Continuous')**

## Querying the Boot Sequence

Function

Query the boot sequence.

Format

**Get-iBMCBootupSequence -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the boot sequence.

PS C:\> **$Sequence = Get-iBMCBootupSequence -Session $session**   
PS C:\> **$Sequence**   
   
Host BootupSequence   
---- --------------   
10.1.1.2 {Pxe, HDD, Cd, Others}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| BootupSequence | Specifies the system boot sequence. The value can be any sequence of **Hdd**, **Cd**, **Pxe**, and **Others**. |

## Setting the Boot Sequence

Function

Set the boot sequence.

Format

**Set-iBMCBootupSequence -Session** *<$session>* **-BootSequence** *<$BootSequence>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<$BootSequence>* | Specifies the system boot sequence. This parameter is mandatory. | The value can be any sequence of **Hdd**, **Cd**, **Pxe**, and **Others**. |

Usage Guidelines

You can set the parameters of multiple services concurrently.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Set the boot sequence.

PS C:\> **$BootSequence = @(@('Pxe', 'Hdd', 'Cd', 'Others'), @('Cd', 'Pxe', 'Hdd', 'Others'))**   
PS C:\> **Set-iBMCBootupSequence -Session $session-BootSequence $BootSequence**

## Exporting the BIOS Configuration

Function

This command is used to export the BIOS configuration.

Format

**Export-iBMCBIOSSetting -Session** *<$session>* **-DestFilePath** *<DestFilePath>* **-SecureEnabled**

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | (Mandatory) Session | - |
| *<DestFilePath>* | (Mandatory) Path for exporting the configuration file | * Using a temporary directory of the iBMC is used: enter **/tmp**/*File name*. * Using a remote directory:   + - When **-SecureEnabled** is configured, the dialog box for entering the user name and password of the remote protocol is displayed. The format of the file path is *File transfer protocol://Remote server IP address/Directory/File name*.     - When **-SecureEnabled** is not configured, the format of the file path is *File transfer protocol://User name:Password@Remote server IP address/Directory/File name*.     - File transfer protocols include HTTPS, SCP, SFTP, CIFS, and NFS. |
| -SecureEnabled | (Optional) Whether to enable the interactive function. The function is enabled when this parameter is added to the command. You are advised to enable this function when using a remote directory to ensure operation security. | - |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Export the BIOS configuration file to the **tmp** directory of the iBMC.

PS C:\> **$Tasks = Export-iBMCBIOSSetting -Session $session -DestFilePath '/tmp/bios.xml'**   
PS C:\> **$Tasks**   
   
Host : 10.1.1.2   
Id : 4   
Name : Export Config File Task   
ActivityName : [10.1.1.2] Export Config File Task   
TaskState : Completed   
StartTime : 2018-11-14T17:52:01+08:00   
EndTime : 2018-11-14T17:53:20+08:00   
TaskStatus : OK   
TaskPercent : 100%

Output Description

Output description

| Output | Description |
| --- | --- |
| Host | Host name |
| Id | Task ID |
| Name | Task name |
| ActivityName | Name of the task activity |
| TaskState | Task execution status |
| StartTime | Start time of a task |
| EndTime | End time of a task |
| TaskStatus | Return status of the connection information |
| TaskPercent | Task progress |

## Importing the BIOS Configuration

Function

Import the BIOS configuration.

Format

**Import-iBMCBIOSSetting -Session** *<$session>* **-ConfigFilePath** *<ConfigFilePath>* **-SecureEnabled**

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<ConfigFilePath>* | Specifies the import path. This parameter is mandatory. | * If a local directory is used, the value must be in *Directory*\*File name* or *\\Server IP address*/*Directory*/*File name* format. * If a local temporary directory of the iBMC is used, the directory must be the **/tmp** directory and a file name must be specified. * Using a remote directory:   + - When **-SecureEnabled** is configured, the dialog box for entering the user name and password of the remote protocol is displayed. The format of the file path is *File transfer protocol://Remote server IP address/Directory/File name*.     - When **-SecureEnabled** is not configured, the format of the file path is *File transfer protocol://User name:Password@Remote server IP address/Directory/File name*.     - File transfer protocols include HTTPS, SCP, SFTP, CIFS, and NFS. |
| -SecureEnabled | (Optional) Whether to enable the interactive function. The function is enabled when this parameter is added to the command. You are advised to enable this function when using a remote directory to ensure operation security. | - |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Import the BIOS configuration.

PS C:\> **$Tasks = Import-iBMCBIOSSetting -Session $session-ConfigFilePath 'C:\10.10.10.2.xml'**   
PS C:\> **$Tasks**   
   
Host : 10.1.1.2   
Id : 2   
Name : Import Config File Task   
ActivityName : [10.1.1.3] Import Config File Task   
TaskState : Completed   
StartTime : 2018-11-14T17:54:54+08:00   
EndTime : 2018-11-14T17:56:06+08:00   
TaskStatus : OK   
TaskPercent : 100%

Output Description

Output description

| Output | Description |
| --- | --- |
| Host | Host name |
| Id | Task ID |
| Name | Task name |
| ActivityName | Name of the task activity |
| TaskState | Task execution status |
| StartTime | Start time of a task |
| EndTime | End time of a task |
| TaskStatus | Return status of the connection information |
| TaskPercent | Task progress |

## Restoring the Factory Settings

Function

Restore the factory settings.



This operation may affect the proper running of the server. Therefore, exercise caution when performing this operation.

Format

**Restore-iBMCFactorySetting -Session***<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

This function is supported only for iBMC 3.08 or later.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Restore the factory settings.

PS C:\> **Restore-iBMCFactorySetting -Session $session**

## Restoring the Default BIOS Settings

Function

Restore the default BIOS settings.



This operation may affect the proper running of the server. Therefore, exercise caution when performing this operation.

Format

**Reset-iBMCBIOSSetting -Session***<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

* Only V5 servers support this function.
* This function takes effect only after the system is restarted.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Restore the default BIOS settings.

PS C:\> **Reset-iBMCBIOSSetting -Session $session**

## Querying BIOS Settings

Function

Query BIOS settings.

Format

**Get-iBMCBIOSSetting -Session** *<$session>* **-Attributes** *<Attributes>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<Attributes>* | Specifies the BIOS attribute value, which is a character string array. This parameter is optional, and fuzzy match is supported. Use commas (,) to separate filter criteria in the same group. | - |

Usage Guidelines

Only V5 servers support this function.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query BIOS settings.

PS C:\> **$System = Get-iBMCBIOSSetting -Session $session -Attributes "TcoTimeout"**   
PS C:\> **$System**   
   
Host : 192.168.1.1   
Id : Bios   
Name : BIOS Configuration Current Settings   
AttributeRegistry : BiosAttributeRegistry.7.6.2   
Attributes : @{TcoTimeout=2}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the name of a host. |
| Id | Specifies the ID of a BIOS resource. |
| Name | Specifies the name of a BIOS resource. |
| AttributeRegistry | Specifies the ID of a registered BIOS resource. |
| Attributes | Specifies the list of BIOS attributes that are currently set. |

## Modifying BIOS Settings

Function

Modify the BIOS resource attributes.

Format

**Set-iBMCBIOSSetting -Session** *<$session>* **-Attributes** *<Attributes>* **-files** *<filePath>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<Attributes>* | Specifies the BIOS attribute value, which is a character string array. This parameter is optional. | - |
| *<filePath>* | Specifies the local JSON file path, which is a character string array. This parameter is optional. | - |

Usage Guidelines

* Only V5 servers support this function.
* After modifying the BIOS settings, you need to restart the server for the modifications to take effect.
* The following is an example of the JSON file format:

{   
 "Attributes": {   
 "BMCWDTTimeout": 25   
 }   
}

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Modify the BIOS resource attributes.

PS C:\> **Set-iBMCBIOSSetting -Session $session** **-Attributes @(@{"BMCWDTTimeout"=18})**   
   
Host : 192.168.1.1   
Id : Settings   
Name : BIOS Configuration Pending Settings   
AttributeRegistry : BiosAttributeRegistry.7.6.2   
Attributes : @{BMCWDTTimeout=20}   
Oem : @{xFusion=}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the name of a host. |
| Id | Specifies the ID of a BIOS resource. |
| Name | Specifies the name of a BIOS resource. |
| AttributeRegistry | Specifies the ID of a registered BIOS resource. |
| Oem | Specifies the custom attributes. |

## Querying the NTP Resource Information

Function

Query the NTP resource information.

Format

**Get-iBMCNTPSetting -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the NTP resource information.

PS C:\> **Get-iBMCNTPSetting -Session $session**   
   
Host : 10.1.1.2   
ServiceEnabled : True   
PreferredNtpServer : pre.xfusion.com   
AlternateNtpServer : alt.xfusion.com   
NtpAddressOrigin : Static   
MinPollingInterval : 10   
MaxPollingInterval : 12   
ServerAuthenticationEnabled : False

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| ServiceEnabled | Specifies whether to enable NTP. |
| PreferredNtpServer | Specifies the preferred NTP server. |
| AlternateNtpServer | Specifies the alternate NTP server. |
| NtpAddressOrigin | Specifies the NTP mode.   * Static * IPv4 * IPv6 |
| MinPollingInterval | Specifies the minimum polling interval. |
| MaxPollingInterval | Specifies the maximum polling interval. |
| ServerAuthenticationEnabled | Specifies the server authentication enabling status. |

## Setting NTP Resource Attributes

Function

Set NTP resource attributes.

Format

**Set-iBMCNTPSetting -Session***<$session>* **-ServiceEnabled** *<ServiceEnabled>* **-PreferredNtpServer** *<PreferredNtpServer>* **-AlternateNtpServer***<AlternateNtpServer>* **-NtpAddressOrigin** *<NtpAddressOrigin>* **-MinPollingInterval** *<MinPollingInterval>* **-MaxPollingInterval** *<MaxPollingInterval>* **-ServerAuthenticationEnabled** *<ServerAuthenticationEnabled>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<ServiceEnabled>* | Specifies whether to enable the NTP service. This parameter is optional. | * **$true** or **1**: enables the NTP function. * **$false** or **0**: disables the NTP function. |
| *<PreferredNtpServer>* | Specifies the preferred NTP server address. This parameter is optional. | * IPv4 address * IPv6 address * Domain name (non-Chinese character string consisting of 1 to 67 characters) |
| *<AlternateNtpServer>* | Specifies the alternate NTP server address. This parameter is optional. | * IPv4 address * IPv6 address * Domain name (non-Chinese character string consisting of 1 to 67 characters) |
| *<NtpAddressOrigin>* | Specifies the address mode. This parameter is optional. | * **Static**: configures the address manually. * **IPv4**: obtains the address from the IPv4 protocol stack automatically. * **IPv6**: obtains the address from the IPv6 protocol stack automatically. |
| *<MinPollingInterval>* | Specifies the minimum polling interval. This parameter is optional. | Value range: The value ranges from 3 to 17 and cannot be greater than the maximum polling interval. |
| *<MaxPollingInterval>* | Specifies the maximum polling interval. This parameter is optional. | Value range: The value ranges from 3 to 17 and cannot be smaller than the minimum polling interval. |
| *<ServerAuthenticationEnabled>* | Specifies the server authentication enabling status. This parameter is optional. | * **$true** or **1**: enables server authentication. * **$false** or **0**: disables server authentication. |

Usage Guidelines

You can set the parameters of multiple services concurrently.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Set NTP resource attributes.

PS C:\> **Set-iBMCNTPSetting -Session $session -ServiceEnabled $true -PreferredNtpServer 'pre.xfusion.com' -AlternateNtpServer 'alt.xfusion.com' ` -NtpAddressOrigin Static -ServerAuthenticationEnabled $false ` -MinPollingInterval 10 -MaxPollingInterval 12**

## Importing the NTP Group Key

Function

Import the NTP group key.

Format

**Import-iBMCNTPGroupKey -Session***<$session>* **-KeyFileUri** *<KeyFileUri>* **-SecureEnabled**

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<KeyFileUri>* | Specifies the path of the key file. This parameter is mandatory. | * If a local directory is used, the value must be in *Directory*\*File name* or *\\Server IP address*/*Directory*/*File name* format. * If a local temporary directory of the iBMC is used, the directory must be the **/tmp** directory and a file name must be specified. * Using a remote directory:   + - When **-SecureEnabled** is configured, the dialog box for entering the user name and password of the remote protocol is displayed. The format of the file path is *File transfer protocol://Remote server IP address/Directory/File name*.     - When **-SecureEnabled** is not configured, the format of the file path is *File transfer protocol://User name:Password@Remote server IP address/Directory/File name*.     - File transfer protocols include HTTPS, SCP, SFTP, CIFS, and NFS. |
| -SecureEnabled | (Optional) Whether to enable the interactive function. The function is enabled when this parameter is added to the command. You are advised to enable this function when using a remote directory to ensure operation security. | - |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Import the NTP group key.

PS C:\> **Import-iBMCNTPGroupKey -Session $session -KeyFileUri "E:\ntp.keys"**   
   
Host : 10.1.1.2   
Id : 1   
Name : ntp certificate import   
ActivityName : [10.1.1.2] ntp certificate import   
TaskState : Completed   
StartTime : 2018-12-21T05:51:46+00:00   
EndTime : 2018-12-21T05:51:49+00:00   
TaskStatus : OK   
TaskPercent : 100%

## Querying the SMTP Notification List

Function

Query the SMTP notification list.

Format

**Get-iBMCSMTPRecipients -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the SMTP notification list.

PS C:\> **$recipients = Get-iBMCSMTPRecipients -Session $session**   
PS C:\> **$recipients**   
   
Host : 10.1.1.2   
MemberId : 0   
Enabled : False   
EmailAddress :   
Description :   
   
Host : 10.1.1.2   
MemberId : 1   
Enabled : False   
EmailAddress :   
Description :   
   
Host : 10.1.1.2   
MemberId : 2   
Enabled : False   
EmailAddress :   
Description :   
Description :   
   
Host : 10.1.1.2   
MemberId : 3   
Enabled : False   
EmailAddress :   
Description :

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| MemberId | Specifies the ID. |
| Enabled | Specifies the enabling status. |
| EmailAddress | Specifies the receiver email address. |
| Description | Specifies the description of the sent content. |

## Setting the SMTP Notification Object

Function

Set the SMTP notification object.

Format

**Set-iBMCSMTPRecipient -Session***<$session>* **-MemberId** *<MemberId>* **-Enabled** *<Enabled>* **-EmailAddress** *<EmailAddress>* -**Description** *<Description>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<MemberId>* | Specifies the ID, which uniquely identifies the notification object in the notification object list. This parameter is mandatory. | Only integers are supported.  The value ranges from 0 to 3. |
| *<Enabled>* | Specifies the email enabling status. This parameter is optional. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<EmailAddress>* | Specifies the receiver email address. This parameter is optional. | - |
| *<Description>* | Specifies the description of the sent content. This parameter is optional. | - |

Usage Guidelines

You can set multiple attributes concurrently.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Set the SMTP notification object.

PS C:\> **Set-iBMCSMTPRecipient -Session $session -MemberId 1 -Enabled $true -EmailAddress r2@xfusion.com -Description 'desc'**

## Querying the SMTP Resource Information

Function

Query the SMTP resource information.

Format

**Get-iBMCSMTPSetting -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the SMTP resource information.

PS C:\> **Get-iBMCSMTPSetting -Session $session**   
   
Host : 10.1.1.2   
ServiceEnabled : True   
ServerAddress : smtp.xfusion.com   
TLSEnabled : True   
AnonymousLoginEnabled : False   
SenderUserName : smtp-sender   
SenderAddress : smtp-sender@xfusion.com   
EmailSubject : Server Alert   
EmailSubjectContains : {HostName, BoardSN, ProductAssetTag}   
AlarmSeverity : Major

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| ServiceEnabled | Specifies the SMTP service enabling status. |
| ServerAddress | Specifies the SMTP server address. |
| TLSEnabled | Specifies whether to enable the TLS function. |
| AnonymousLoginEnabled | Specifies whether to use the anonymous mode. |
| SenderUserName | Specifies the user name of the sender. |
| SenderAddress | Specifies the sender email address. |
| EmailSubject | Specifies the email subject. |
| EmailSubjectContains | Specifies the additional information of the subject.   * HostName * BoardSN * ProductAssetTag |
| AlarmSeverity | Specifies the alarm sending level. |

## Setting SMTP Resource Attributes

Function

Set SMTP resource attributes.

Format

**Set-iBMCSMTPSetting -Session***<$session>* **-ServiceEnabled** *<ServiceEnabled>* **-ServerAddress** *<ServerAddress>* **-TLSEnabled** *<TLSEnabled>* **-AnonymousLoginEnabled** *<AnonymousLoginEnabled>* **-SenderUserName** *<SenderUserName>* **-SenderAddress** *<SenderAddress>* **-SenderPassword** *<SenderPassword>* **-EmailSubject** *<EmailSubject>* **-EmailSubjectContains** *<EmailSubjectContains>* **-AlarmSeverity** *<AlarmSeverity>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<ServiceEnabled>* | Specifies the SMTP service enabling status. This parameter is optional. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<ServerAddress>* | Specifies the SMTP server address. This parameter is optional. | - |
| *<TLSEnabled>* | Specifies whether to enable the TLS function. This parameter is optional. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<AnonymousLoginEnabled>* | Specifies whether to use the anonymous mode. This parameter is optional. | * **$true** or **1**: anonymous * **$false** or **0**: non-anonymous |
| *<SenderUserName>* | Specifies the user name of the sender. This parameter is optional. | - |
| *<SenderAddress>* | Specifies the email address of the sender. This parameter is optional. | - |
| *<SenderPassword>* | Specifies the password of the sender. This parameter is optional. | - |
| *<EmailSubject>* | Specifies the email subject. This parameter is optional. | - |
| *<EmailSubjectContains>* | Specifies the additional information of the subject. This parameter is optional. | You can select one or more of the following additional items:   * **HostName**: indicates the host name. * **BoardSN**: indicates the board serial number. * **ProductAssetTag**: indicates the product asset tag. |
| *<AlarmSeverity>* | Specifies the alarm sending level. This parameter is optional. | * Critical * Major * Minor * Normal |

Usage Guidelines

You can set the parameters of multiple services concurrently.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Set SMTP resource attributes.

PS C:\> **$pwd = ConvertTo-SecureString -String "pwd12#$%^" -AsPlainText -Force**   
PS C:\> **$ServerIdentifer = ,@('HostName', 'BoardSN')**   
PS C:\> **Set-iBMCSMTPSetting -Session $session -ServiceEnabled $false -ServerAddress smtp.xfusion.com ` -TLSEnabled $false -AnonymousLoginEnabled $false ` -SenderUserName 'xFusion-iBMC' -SenderAddress "powershell@xfusion.com" -SenderPassword $pwd ` -EmailSubject 'iBMC Alarm Notification' -EmailSubjectContains $ServerIdentifer ` -AlarmSeverity Critical**

## Querying the SNMP Resource Information

Function

Query the SNMP resource information.

Format

**Get-iBMCSNMPSetting -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the SNMP resource information.

PS C:\> **Get-iBMCSNMPSetting -Session $session**   
   
Host : 10.1.1.2   
SnmpV1Enabled : False   
SnmpV2CEnabled : False   
SnmpV3Enabled : True   
LongPasswordEnabled : True   
RWCommunityEnabled : True   
SnmpV3AuthProtocol : MD5   
SnmpV3PrivProtocol : DES

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| SnmpV1Enabled | Specifies the SNMPv1 enabling status. |
| SnmpV2CEnabled | Specifies the SNMPv2c enabling status. |
| SnmpV3Enabled | Specifies the SNMPv3 enabling status. |
| LongPasswordEnabled | Specifies the long password enabling status. |
| RWCommunityEnabled | Specifies the read-write community name enabling status. |
| SnmpV3AuthProtocol | Specifies the SNMPv3 authentication algorithm. |
| SnmpV3PrivProtocol | Specifies the SNMPv3 encryption algorithm. |

## Setting SNMP Resource Attributes

Function

Set SNMP resource attributes.

Format

**Set-iBMCSNMPSetting -Session** *<$session>* **-SnmpV1Enabled** *<SnmpV1Enabled>* **-SnmpV2CEnabled** *<SnmpV2CEnabled>* **-LongPasswordEnabled** *<LongPasswordEnabled>* **-RWCommunityEnabled** *<RWCommunityEnabled>* **-ReadOnlyCommunity** *<ReadOnlyCommunity>* **-ReadWriteCommunity** *<ReadWriteCommunity>* **-SnmpV3AuthProtocol** *<SnmpV3AuthProtocol>* **-SnmpV3PrivProtocol** *<SnmpV3PrivProtocol>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| <$session> | Specifies the session. This parameter is mandatory. | - |
| *<SnmpV1Enabled>* | Specifies the SNMPv1 enabling status. This parameter is optional. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<SnmpV2CEnabled>* | Specifies the SNMPv2c enabling status. This parameter is optional. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<LongPasswordEnabled>* | Specifies the long password enabling status. This parameter is optional. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<RWCommunityEnabled>* | Specifies the read-write community name enabling status. This parameter is optional. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<ReadOnlyCommunity>* | Specifies the read-only community name. This parameter is optional. | Value requirements:   * The value cannot contain spaces. * The length is 1 to 32 bytes. After the long password is enabled, the length is 16 to 32 bytes. * The value must contain at least two characters different from the characters in the password set last time. * The read-only community name and read-write community name cannot be the same. * When the password complexity check function is enabled, the minimum length is 8 bytes. The password must contain at least two types of the following characters: lowercase letters, uppercase letters, digits, and special characters. |
| *<ReadWriteCommunity>* | Specifies the read-write community name. This parameter is optional. | Value requirements:   * The value cannot contain spaces. * The length is 1 to 32 bytes. After the long password is enabled, the length is 16 to 32 bytes. * The value must contain at least two characters different from the characters in the password set last time. * The read-only community name and read-write community name cannot be the same. * When the password complexity check function is enabled, the minimum length is 8 bytes. The password must contain at least two types of the following characters: lowercase letters, uppercase letters, digits, and special characters. |
| *<SnmpV3AuthProtocol>* | Specifies the SNMPv3 authentication algorithm. This parameter is optional. | * MD5 * SHA |
| *<SnmpV3PrivProtocol>* | Specifies the SNMPv3 encryption algorithm. This parameter is optional. | * DES * AES |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Set SNMP resource attributes.

PS C:\> **$ReadOnlyCommunity = ConvertTo-SecureString -String "SomeP@ssw0rd1" -AsPlainText -Force**   
PS C:\> **$ReadWriteCommunity = ConvertTo-SecureString -String "SomeP@ssw0rd2" -AsPlainText -Force**   
PS C:\> **Set-iBMCSNMPSetting -Session $session -SnmpV1Enabled $false -SnmpV2CEnabled $false ` -LongPasswordEnabled $true -RWCommunityEnabled $true ` -ReadOnlyCommunity $ReadOnlyCommunity -ReadWriteCommunity $ReadWriteCommunity ` -SnmpV3AuthProtocol MD5 -SnmpV3PrivProtocol DES**

## Querying the SNMP Trap Server

Function

Query the SNMP Trap server.

Format

**Get-iBMCSNMPTrapServer -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the SNMP Trap notification list.

PS C:\> **$tasks = Get-iBMCSNMPTrapServer -Session $sessions**   
PS C:\> **$tasks**   
   
Host : 10.1.1.2   
MemberId : 0   
BobEnabled : False   
Enabled : False   
TrapServerAddress :   
TrapServerPort : 300   
   
Host : 10.1.1.2   
MemberId : 1   
BobEnabled : False   
Enabled : True   
TrapServerAddress : 10.10.10.3   
TrapServerPort : 310   
   
Host : 10.1.1.2   
MemberId : 2   
BobEnabled : False   
Enabled : False   
TrapServerAddress : 10.10.10.4   
TrapServerPort : 163   
   
Host : 10.1.1.2   
MemberId : 3   
BobEnabled : True   
Enabled : True   
TrapServerAddress : 10.10.10.2   
TrapServerPort : 202

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| MemberId | Specifies the array ID. |
| BobEnabled | Specifies the enabling status of in-band forwarding. |
| Enabled | Specifies the server enabling status. |
| TrapServerAddress | Specifies the server address. |
| TrapServerPort | Specifies the server port number. |

## Setting the SNMP Trap Server

Function

Set the SNMP Trap server.

Format

**Set-iBMCSNMPTrapServer -Session***<$session>* **-MemberId** *<MemberId>* **-Enabled** *<Enabled>* **-TrapServerAddress** <TrapServerAddress> **-TrapServerPort** <TrapServerPort>

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<MemberId>* | Specifies the array ID, which uniquely identifies the notification object in the notification object list. This parameter is mandatory. | Only integers are supported.  The value ranges from 0 to 3. |
| *<Enabled>* | Specifies the server enabling status. This parameter is optional. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<TrapServerAddress>* | Specifies the Trap server address. This parameter is optional. | The value is an IPv4 or IPv6 address or a domain name character string. |
| *<TrapServerPort>* | Specifies the SNMP Trap port number. This parameter is optional. | 1 to 65535 |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Set the SNMP Trap notification object.

PS C:\> **Set-iBMCSNMPTrapServer -Session $session -MemberId 1 -Enabled $true -TrapServerAddress 10.10.10.3 -TrapServerPort 1024**

## Querying the SNMP Trap Resource Information

Function

Query the SNMP Trap resource information.

Format

**Get-iBMCSNMPTrapSetting -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the SNMP Trap resource information.

PS C:\> **Get-iBMCSNMPTrapSetting -Session $session**   
   
Host : 10.1.1.2   
ServiceEnabled : True   
TrapVersion : V2C   
TrapV3User : UserName   
TrapMode : EventCode   
TrapServerIdentity : BoardSN   
AlarmSeverity : Critical

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| ServiceEnabled | Specifies the Trap function enabling status. |
| TrapVersion | Specifies the SNMP Trap version information. |
| TrapV3User | Specifies the SNMP Trap V3 user name. |
| TrapMode | Specifies the SNMP Trap reporting mode.   * OID * EventCode * PreciseAlarm |
| TrapServerIdentity | Specifies the host ID. |
| AlarmSeverity | Specifies the alarm sending level. |

## Setting SNMP Trap Resource Attributes

Function

Set SNMP Trap resource attributes.

Format

**Set-iBMCSNMPTrapSetting -Session** *<$session>* **-ServiceEnabled** *<ServiceEnabled>* **-TrapVersion** *<TrapVersion>* **-TrapV3User** *<TrapV3User>* **-TrapMode** *<TrapMode>* **-TrapServerIdentity** *<TrapServerIdentity>* **-CommunityName** *<CommunityName>* **-AlarmSeverity** *<AlarmSeverity>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<ServiceEnabled>* | Specifies the Trap function enabling status. This parameter is optional. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<TrapVersion>* | Specifies the SNMP Trap version. This parameter is optional. | * V1 * V2C * V3 |
| *<TrapV3User>* | Specifies the SNMP Trap V3 user name. This parameter is optional. | The value is an existing local user name.  This parameter is valid only when **TrapVersion** is set to **V3**. |
| *<TrapMode>* | Specifies the SNMP Trap reporting mode. This parameter is optional. | * **OID**: indicates the OID mode. * **EventCode**: indicates the event code mode. * **PreciseAlarm**: indicates the precise alarm mode. |
| *<TrapServerIdentity>* | Specifies the user name of the sender. This parameter is optional.  This parameter is mandatory when **TrapMode** is set to **OID** or **PreciseAlarm**. | * BoardSN * ProductAssetTag * HostName |
| *<CommunityName>* | Specifies the SNMP Trap community name. This parameter is optional. | Value requirements:   * The Trap community name cannot be set when the Trap version is V3. * Password complexity check:   When this function is enabled, the value must be a combination of at least two types of the following characters: lowercase letters, uppercase letters, digits, and special characters. The length must be 8 to 18 bytes. The value must contain at least two characters different from the characters in the password set last time.  When this function is disabled, the length must be 1 to 18 bytes.   * The value cannot contain spaces.   NOTE  If **TrapV3User** is set, the community name is invalid. |
| *<AlarmSeverity>* | Specifies the alarm sending level. This parameter is optional. | * Critical * Major * Minor * Normal |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Set SNMP Trap resource attributes.

PS C:\> **$CommunityName = ConvertTo-SecureString -String "SomeP@ssw0rd" -AsPlainText -Force**   
PS C:\> **Set-iBMCSNMPTrapSetting -Session $session -ServiceEnabled $true -TrapVersion V2C ` -TrapV3User chajian -TrapMode EventCode -TrapServerIdentity BoardSN ` -CommunityName $CommunityName -AlarmSeverity Critical**

## Querying the Syslog Notification List

Function

Query the Syslog notification list.

Format

**Get-iBMCSyslogServer -Session***<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the Syslog notification list.

PS C:\> **$Servers = Get-iBMCSyslogServer -Session $session**   
PS C:\> **$Servers**   
   
Host : 10.1.1.2   
MemberId : 0   
Enabled : False   
Address :   
Port : 0   
LogType : {OperationLog, SecurityLog, EventLog}   
   
Host : 10.1.1.2   
MemberId : 1   
Enabled : False   
Address :   
Port : 0   
LogType : {OperationLog, SecurityLog, EventLog}   
   
Host : 10.1.1.2   
MemberId : 2   
Enabled : False   
Address :   
Port : 0   
LogType : {OperationLog, SecurityLog, EventLog}   
   
Host : 10.1.1.2   
MemberId : 3   
Enabled : False   
Address :   
Port : 0   
LogType : {OperationLog, SecurityLog, EventLog}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| MemberId | Specifies the array ID. |
| Enabled | Specifies the server enabling status. |
| Address | Specifies the server address. |
| Port | Specifies the Syslog server port number. |
| LogType | Specifies the type of reported logs. |

## Setting the Syslog Notification Object

Function

Set the Syslog notification object.

Format

**Set-ibmcSyslogServer -Session** *<$session>* **-MemberId** *<MemberId>* **-Enabled** *<Enabled>* **-Address** *<Address>* **-Port** *<Port>* **-LogType** *<LogType>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<MemberId>* | Specifies the array ID, which uniquely identifies the notification object in the notification object list. This parameter is mandatory. | Only integers are supported.  The value ranges from 0 to 3. |
| *<Enabled>* | Specifies the server enabling status. This parameter is optional. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<Address>* | Specifies the server address. This parameter is optional. | The value is an IPv4 or IPv6 address or a domain name character string. |
| *<Port>* | Specifies the Syslog server port number. This parameter is optional. | 1 to 65535 |
| <LogType> | Specifies the type of reported logs. This parameter is optional. | You can select one or more of the following log types:   * **OperationLog**: indicates that operations logs are reported. * **SecurityLog**: indicates that security logs are reported. * **EventLog**: indicates that event logs are reported. |

Usage Guidelines

You can set the parameters of multiple services concurrently.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Set the Syslog notification object.

PS C:\> **$LogType = ,@("OperationLog", "SecurityLog", "EventLog")**   
PS C:\> **Set-ibmcSyslogServer -Session $session -MemberId 1 -Enabled $true -Address 10.10.10.2 -Port 515 -LogType $LogType**

## Querying the Syslog Resource Information

Function

Query the Syslog resource information.

Format

**Get-iBMCSyslogSetting -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the Syslog resource information.

PS C:\> **$syslog = Get-iBMCSyslogSetting -Session $session**   
PS C:\> **$syslog**   
   
Host : 10.1.1.2   
ServiceEnabled : True   
ServerIdentitySource : BoardSN   
AlarmSeverity : Normal   
TransmissionProtocol : UDP

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| ServiceEnabled | Specifies the Syslog service enabling status. |
| ServerIdentitySource | Specifies the Syslog host ID. |
| AlarmSeverity | Specifies the alarm severity.   * Normal * Minor * Major * Critical |
| TransmissionProtocol | Specifies the transmission protocol type.   * UDP * TCP * TLS |

## Setting the Syslog Resource Information

Function

Set the Syslog resource information.

Format

**Set-iBMCSyslogSetting -Session** *<$session>* **-ServiceEnabled** *<ServiceEnabled>* **-ServerIdentitySource** *<ServerIdentitySource>* **-AlarmSeverity** *<AlarmSeverity>* **-TransmissionProtocol**

*<TransmissionProtocol>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<ServiceEnabled>* | Specifies the Syslog service enabling status. This parameter is optional. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<ServerIdentitySource>* | Indicates the Syslog host ID. This parameter is optional. | * **BoardSN**: indicates the board serial number. * **ProductAssetTag**: indicates the product asset tag. * **HostName**: indicates the host name. |
| *<AlarmSeverity>* | Specifies the alarm severity. This parameter is optional. | * Critical * Major * Minor * Normal |
| *<TransmissionProtocol>* | Specifies the transmission protocol type. This parameter is optional. | * UDP * TCP * TLS |

Usage Guidelines

You can set the parameters of multiple services concurrently.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Set the Syslog resource information.

PS C:\> **Set-iBMCSyslogSetting -Session $session -ServiceEnabled $true -ServerIdentitySource HostName ` -AlarmSeverity Major -TransmissionProtocol UDP**

## Querying the In-Band Firmware Version

Function

Query the in-band firmware version.

Format

**Get-iBMCInbandFirmware** **-Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the in-band firmware version.

PS C:\> **Get-iBMCInbandFirmware -Session $session**   
   
 Host : 10.1.1.2   
 SR430C-M 1G (SAS3108)@[RAID Card1] : 4.270.00-4382   
 LOM (X722)@[LOM] : 3.33 0x80000f09 255.65535.255   
 SPService : @{APPVersion=1.09; OSVersion=1.09; DataVersion=1.09}

## Querying the Out-of-Band Firmware Version

Function

Query the out-of-band firmware version.

Format

**Get-iBMCOutbandFirmware -Session***<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the out-of-band firmware version.

PS C:\> **Get-iBMCOutbandFirmware -Session $session**   
   
Host : 10.1.1.2   
ActiveBMC : 3.18   
BackupBMC : 3.18   
Bios : 0.81   
MainBoardCPLD : 2.02   
chassisDiskBP1CPLD : 1.10

## Upgrading the In-Band Firmware

Function

Upgrade the in-band firmware.

Format

**Update-iBMCInbandFirmware -Session** *<$session>***-Type** *<Type>* **-FileUri** *<FileUri>* **-SignalFileUri** *<SignalFileUri>* **-UpgradeMode** *<UpgradeMode>* **-SecureEnabled**

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<Type>* | Specifies the firmware type. This parameter is mandatory. | * **Firmware**: RAID controller card and NIC * **SP**: Smart Provisioning |
| *<FileUri>* | Specifies the path of the upgrade file on the server. This parameter is mandatory.  NOTE   * The firmware upgrade file is in .zip format. * The Smart Provisioning upgrade file is in .ISO format. | * The directory where the upgrade file is stored cannot contain the following special characters: ||,;&&$|>>><. * Only remote directories are supported.   + - When **-SecureEnabled** is configured, the dialog box for entering the user name and password of the remote protocol is displayed. The format of the file path is *File transfer protocol://Remote server IP address/Directory/File name*.     - When **-SecureEnabled** is not configured, the format of the file path is *File transfer protocol://User name:Password@Remote server IP address/Directory/File name*. * If the value of **Type** is **Firmware**, the HTTPS, SFTP, NFS, CIFS, SCP, and FILE file transfer protocols are supported. * If the value of **Type** is **SP**, only the NFS and CIFS file transfer protocols are supported. |
| *<SignalFileUri>* | Specifies the digital signature (in .asc format) of the upgrade package.  If the value of **Type** is **Firmware**, this parameter is mandatory. | * The directory where the upgrade file is stored cannot contain the following special characters: ||,;&&$|>>><. * The HTTPS, SFTP, NFS, CIFS, SCP, and FILE file transfer protocols are supported. |
| *<UpgradeMode>* | Specifies the upgrade mode. This parameter is optional. | * Auto * Full * Recover * APP * Driver |
| -SecureEnabled | (Optional) Whether to enable the interactive function. The function is enabled when this parameter is added to the command. You are advised to enable this function when using a remote directory to ensure operation security. | - |

Usage Guidelines

None

Example

# Use a file in a remote directory to upgrade Smart Provisioning, enable Smart Provisioning, and restart the server.

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Upgrade the in-band firmware.

PS C:\> **Update-iBMCInbandFirmware -Session $session -Type Firmware `**   
  **-FileUri "nfs://10.10.10.2/data/nfs/NIC(X722)-Electrical-05022FTM-FW(3.33).zip" `**   
  **-SignalFileUri "nfs://10.10.10.2/data/nfs/NIC(X722)-Electrical-05022FTM-FW(3.33).zip.asc" `**   
  **-UpgradeMode Recover**

1. Enable Smart Provisioning.

PS C:\> **Set-iBMCSPService -Session $session -StartEnabled $true -SysRestartDelaySeconds 60**

1. Restart the server.

PS C:\> **Set-iBMCServerPower -Session $session -ResetType ForceRestart**

1. Query the configuration result resource of the Smart Provisioning service.

PS C:\> **$Result = Get-iBMCSPTaskResult -Session $session**   
PS C:\> **$Result**   
   
Host : 10.1.1.2   
Id : 1   
Name : SP Result   
Status : Finished   
OSInstall :   
Clone :   
Recover :   
Upgrade : @{Progress=Successful; Detail=System.Object[]}

1. Check whether the firmware versions are upgraded to the target versions.

PS C:\> **Get-iBMCInbandFirmware -Session $session**   
   
Host : 10.1.1.2   
SR430C-M 1G (SAS3108)@[RAID Card1] : 4.270.00-4382   
LOM (X722)@[LOM] : 3.33 0x80000f09 255.65535.255   
SPService : @{APPVersion=1.16; OSVersion=1.16; DataVersion=1.16}

# Use a file in a remote directory to upgrade Smart Provisioning.

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Upgrade Smart Provisioning.

PS C:\> **Update-iBMCInbandFirmware -Session $session -Type SP `**   
 **-FileUri "nfs://10.10.10.2/data/nfs/Firmware.ISO `**   
 **-UpgradeMode Recover**

1. Check whether the firmware versions are upgraded to the target versions.

PS C:\> **Get-iBMCInbandFirmware $session**   
   
Host : 10.1.1.2   
SR430C-M 1G (SAS3108)@[RAID Card1] : 4.270.00-4382   
LOM (X722)@[LOM] : 3.33 0x80000f09 255.65535.255   
SPService : @{APPVersion=1.16; OSVersion=1.16; DataVersion=1.16}

## Upgrading the Out-of-Band Firmware

Function

Upgrade the out-of-band firmware.

Format

**Update-iBMCOutbandFirmware -Session** *<$session>* **-FileUri** *<FileUri>* **-SecureEnabled**

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| <FileUri> | Specifies the path of the upgrade file on the server. This parameter is mandatory. | The file URL cannot exceed 256 characters.   * If a local directory is used, the value must be in *Directory*\*File name* or *\\Server IP address*\*Directory*\*File name* format. * Using a remote directory:   + - When **-SecureEnabled** is configured, the dialog box for entering the user name and password of the remote protocol is displayed. The format of the file path is *File transfer protocol://Remote server IP address/Directory/File name*.     - When **-SecureEnabled** is not configured, the format of the file path is *File transfer protocol://User name:Password@Remote server IP address/Directory/File name*.     - The file transfer protocols include HTTPS, SCP, SFTP, CIFS, TFTP, and NFS, and the protocol name must be in lower case. |
| -SecureEnabled | (Optional) Whether to enable the interactive function. The function is enabled when this parameter is added to the command. You are advised to enable this function when using a remote directory to ensure operation security. | - |

Usage Guidelines

The upgrade takes effect after the BIOS and CPLD are powered off.

Example

# Upgrade the out-of-band firmware using a local file.

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Upgrade the out-of-band firmware.

PS C:\> **Update-iBMCOutbandFirmware -Session $session -FileUri E:\2288H\_V5\_5288\_V5-iBMC-V318.hpm**   
   
Host : 10.1.1.2   
Id : 1   
Name : Upgarde Task   
ActivityName : [10.1.1.2] Upgarde Task   
TaskState : Completed   
StartTime : 2018-11-23T08:57:45+08:00   
EndTime : 2018-11-23T09:01:24+08:00   
TaskStatus : OK   
TaskPercent : 100%

1. Check whether the firmware versions are upgraded to the target versions.

PS C:\> **Get-iBMCOutbandFirmware -Session $session**   
   
Host : 10.1.1.2   
ActiveBMC : 3.18   
BackupBMC : 3.18   
Bios : 0.81   
MainBoardCPLD : 2.02   
chassisDiskBP1CPLD : 1.10

# Upgrade the out-of-band firmware using a file in a remote directory.

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Upgrade the out-of-band firmware.

PS C:\> **Update-iBMCOutbandFirmware -Session $session ` -FileUri nfs://10.10.10.3/data/nfs/2288H\_V5\_5288\_V5-iBMC-V318.hpm**   
   
Host : 10.1.1.2   
Id : 1   
Name : Upgarde Task   
ActivityName : [10.1.1.2] Upgarde Task   
TaskState : Completed   
StartTime : 2018-11-23T08:57:45+08:00   
EndTime : 2018-11-23T09:01:24+08:00   
TaskStatus : OK   
TaskPercent : 100%

1. Check whether the firmware versions are upgraded to the target versions.

PS C:\> **Get-iBMCOutbandFirmware -Session $session**   
   
Host : 10.1.1.2   
ActiveBMC : 3.18   
BackupBMC : 3.18   
Bios : 0.81   
MainBoardCPLD : 2.02   
chassisDiskBP1CPLD : 1.10

## Querying the CPU Health Status

Function

Query the overall CPU health status and component health status.

Format

**Get-iBMCProcessorsHealth -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the overall CPU health status and component health status.

PS C:\> **$health = Get-iBMCProcessorsHealth -Session $session**   
 PS C:\> **$health | fl**   
   
 Host : 10.1.1.2   
 Summary : @{HealthRollup=OK}   
 ID#1 : @{Health=OK; State=Enabled; DeviceLocator=CPU1}   
 ID#2 : @{Health=OK; State=Enabled; DeviceLocator=CPU2}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Summary | Specifies the overall CPU health status. |
| ID#*X* | Specifies the health status of the CPU whose ID is *X*. |

## Querying the RAID Controller Card Health Status

Function

Query the overall RAID controller card health status and component health status.

Format

**Get-iBMCRAIDControllersHealth -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the overall RAID controller card health status and component health status.

PS C:\> **$health = Get-iBMCRAIDControllersHealth -Session $session**   
 PS C:\> $health | fl   
   
 Host : 10.1.1.2   
 Summary : @{HealthRollup=OK}   
 ID#RAIDStorage0 : @{Health=OK; State=Enabled; Name=RAID Card1 Controller}   
 ID#RAIDStorage1 : @{Health=OK; State=Enabled; Name=PCIe Card 5 (RAID) Controller}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Summary | Specifies the overall RAID controller card health status. |
| ID#*X* | Specifies the health status of the RAID controller card whose ID is *X*. |

## Querying the Memory Health Status

Function

Query the overall memory health status and component health status.

Format

**Get-iBMCMemoryHealth -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the overall memory health status and component health status.

PS C:\> **$health = Get-iBMCMemoryHealth -Session $session**   
 PS C:\> **$health**   
   
 Host : 10.1.1.2   
 Summary : @{HealthRollup=OK}   
 ID#mainboardDIMM000 : @{Health=OK; State=Enabled; DeviceLocator=DIMM000}   
 ID#mainboardDIMM010 : @{Health=OK; State=Enabled; DeviceLocator=DIMM010}   
 ID#mainboardDIMM030 : @{Health=OK; State=Enabled; DeviceLocator=DIMM030}   
 ID#mainboardDIMM040 : @{Health=OK; State=Enabled; DeviceLocator=DIMM040}   
 ID#mainboardDIMM100 : @{Health=OK; State=Enabled; DeviceLocator=DIMM100}   
 ID#mainboardDIMM110 : @{Health=OK; State=Enabled; DeviceLocator=DIMM110}   
 ID#mainboardDIMM130 : @{Health=OK; State=Enabled; DeviceLocator=DIMM130}   
 ID#mainboardDIMM140 : @{Health=OK; State=Enabled; DeviceLocator=DIMM140}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Summary | Specifies the overall memory health status. |
| ID#*X* | Specifies the health status of the DIMM whose ID is *X*. |

## Querying the Power Supply Health Status

Function

Query the overall power supply health status and component health status.

Format

**Get-iBMCPowerSuppliesHealth -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the overall power supply health status and component health status.

PS C:\> **$health = Get-iBMCPowerSuppliesHealth -Session $session**   
 PS C:\> $health | fl   
   
 Host : 10.1.1.2   
 Summary : @{HealthRollup=Critical}   
 MemberId#0 : @{Health=OK; State=Enabled; Name=PS1}   
 MemberId#1 : @{Health=Critical; State=Enabled; Name=PS2}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Summary | Specifies the overall power supply health status. |
| MemberId#*X* | Specifies the health status of the PSU whose ID is *X*. |

## Querying the Drive Health Status

Function

Query the overall drive health status and component health status.

Format

**Get-iBMCDrivesHealth -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the overall drive health status and component health status.

PS C:\> **$health = Get-iBMCDrivesHealth -Session $session**   
 PS C:\> **$health | fl**   
   
 Host : 10.1.1.2   
 Summary : @{HealthRollup=OK}   
 ID#HDDPlaneDisk0 : @{Health=OK; State=; Name=Disk0}   
 ID#HDDPlaneDisk1 : @{Health=OK; State=; Name=Disk1}   
 ID#HDDPlaneDisk2 : @{Health=OK; State=; Name=Disk2}   
 ID#HDDPlaneDisk3 : @{Health=OK; State=; Name=Disk3}   
 ID#HDDPlaneDisk4 : @{Health=OK; State=; Name=Disk4}   
 ID#HDDPlaneDisk5 : @{Health=OK; State=; Name=Disk5}   
 ID#HDDPlaneDisk6 : @{Health=OK; State=; Name=Disk6}   
 ID#HDDPlaneDisk7 : @{Health=OK; State=; Name=Disk7}   
 ID#HDDPlaneDisk40 : @{Health=OK; State=; Name=Disk40}   
 ID#HDDPlaneDisk41 : @{Health=OK; State=; Name=Disk41}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Summary | Specifies the overall hard drive health status. |
| ID#*X* | Specifies the health status of the hard drive whose ID is *X*. |

## Querying the NIC Health Status

Function

Query the overall NIC health status and component health status.

Format

**Get-iBMCNetworkAdaptersHealth -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the overall NIC health status and component health status.

PS C:\> **$health = Get-iBMCNetworkAdaptersHealth -Session $session**   
 PS C:\> **$health | fl**   
   
 Host : 10.1.1.2   
 Summary : @{HealthRollup=OK}   
 ID#mainboardLOM : @{Health=OK; State=Enabled; Name=mainboardLOM}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Summary | Specifies the overall NIC health status. |
| ID#*X* | Specifies the health status of the NIC whose ID is *X*. |

## Querying the Fan Health Status

Function

Query the overall fan health status and component health status.

Format

**Get-iBMCFansHealth -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the overall fan health status and component health status.

PS C:\> **$health = Get-iBMCFansHealth -Session $session**   
 PS C:\> **$health**   
   
 Host : 10.1.1.2   
 Summary : @{HealthRollup=OK}   
 MemberId#0 : @{Health=OK; State=Enabled; Name=Fan Module1 Front}   
 MemberId#1 : @{Health=OK; State=Enabled; Name=Fan Module2 Front}   
 MemberId#2 : @{Health=OK; State=Enabled; Name=Fan Module3 Front}   
 MemberId#3 : @{Health=OK; State=Enabled; Name=Fan Module4 Front}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Summary | Specifies the overall fan health status. |
| MemberId#*X* | Specifies the health status of the fan whose ID is *X*. |

## Querying the CPU Information

Function

Query the CPU information.

Format

**Get-iBMCProcessors -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the CPU information.

PS C:\> **$ProcessorsArray = Get-iBMCProcessors -Session $session**   
 PS C:\> **$ProcessorsArray**   
   
 Host : 10.1.1.2   
 Id : 1   
 ProcessorType : CPU   
 ProcessorArchitecture : x86   
 InstructionSet : x86-64   
 Manufacturer : Intel(R) Corporation   
 Model : Intel(R) Xeon(R) Silver 4110 CPU @ 2.10GHz   
 IdentificationRegisters : 54-06-05-00-FF-FB-EB-BF   
 MaxSpeedMHz : 4000   
 TotalCores : 8   
 TotalThreads : 16   
 Socket : 0   
 L1CacheKiB : 512   
 L2CacheKiB : 8192   
 L3CacheKiB : 11264   
 DeviceLocator : CPU1   
 Position : mainboard   
 PartNumber : 41020679   
 Temperature : 30   
 EnabledSetting : True   
 FrequencyMHz : 2100   
 OtherParameters : 64-bit Capable| Multi-Core| Hardware Thread| Execute Protection| Enhanced Virtualization| Power/Performance Control   
 Status : @{State=Enabled; Health=Warning}   
   
 Host : 10.1.1.2   
 Id : 2   
 ProcessorType : CPU   
 ProcessorArchitecture : x86   
 InstructionSet : x86-64   
 Manufacturer : Intel(R) Corporation   
 Model : Intel(R) Xeon(R) Silver 4110 CPU @ 2.10GHz   
 IdentificationRegisters : 54-06-05-00-FF-FB-EB-BF   
 MaxSpeedMHz : 4000   
 TotalCores : 8   
 TotalThreads : 16   
 Socket : 1   
 L1CacheKiB : 512   
 L2CacheKiB : 8192   
 L3CacheKiB : 11264   
 DeviceLocator : CPU2   
 Position : mainboard   
 PartNumber : 41020679   
 Temperature : 31   
 EnabledSetting : True   
 FrequencyMHz : 2100   
 OtherParameters : 64-bit Capable| Multi-Core| Hardware Thread| Execute Protection| Enhanced Virtualization| Power/Performance Control   
 Status : @{State=Enabled; Health=OK}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the CPU resource ID. |
| ProcessorArchitecture | Specifies the CPU resource architecture. The options are as follows:   * x86 * IA-64 * MIPS * OEM |
| InstructionSet | Specifies the CPU resource instruction set. The options are as follows:   * x86 * x86-64 * IA-64 * MIPS32 * MIPS64 * OEM |
| Manufacturer | Specifies the CPU resource manufacturer. |
| Model | Specifies the CPU resource model. |
| IdentificationRegisters | Specifies the CPU resource identification register. |
| MaxSpeedMHz | Specifies the maximum frequency of the CPU resource. |
| TotalCores | Specifies the total number of cores of the CPU resource. |
| TotalThreads | Specifies the total number of threads of the CPU resource. |
| Socket | Specifies the CPU resource slot number. |
| L1CacheKiB | Specifies the L1 cache of the CPU resource. |
| L2CacheKiB | Specifies the L2 cache of the CPU resource. |
| L3CacheKiB | Specifies the L3 cache of the CPU resource. |
| DeviceLocator | Specifies the CPU resource silkscreen. |
| Position | Specifies the CPU resource container. |
| PartNumber | Specifies the CPU resource part number. |
| Temperature | Specifies the CPU resource temperature. |
| EnabledSetting | Specifies the CPU resource enabling status. |
| FrequencyMHz | Specifies the CPU resource frequency. |
| OtherParameters | Specifies other parameters. |
| Status | Specifies the CPU resource status. |

## Querying the RAID Controller Card Information

Function

Query the RAID controller card information.

Format

**Get-iBMCRAIDControllers -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the RAID controller card information.

PS C:\> **$RAID = Get-iBMCRAIDControllers -Session $Session**   
 PS C:\> **$RAID**   
   
Host : 10.1.1.2   
Id : RAIDStorage0   
Name : RAID Card1 Controller   
Description : RAID Controller   
Status : @{State=Enabled; Health=OK}   
SpeedGbps : 12   
FirmwareVersion : 5.010.00-0839   
SupportedDeviceProtocols : {SAS}   
Manufacturer :   
Model : SAS3508   
SupportedRAIDLevels : {RAID0, RAID1, RAID5, RAID6...}   
Mode : Non-RAID   
CachePinnedState : False   
SASAddress : 5505dac310072000   
ConfigurationVersion : 4.1610.00-0149   
MemorySizeMiB : 2048   
MaintainPDFailHistory : True   
CopyBackState : True   
SmarterCopyBackState : True   
JBODState : False   
OOBSupport : True   
CapacitanceName :   
CapacitanceStatus : @{State=Absent; Health=}   
DriverInfo : @{DriverName=; DriverVersion=}   
DDRECCCount : 0   
MinStripeSizeBytes : 65536   
MaxStripeSizeBytes : 1048576   
Drives : {HDDPlaneDisk0, HDDPlaneDisk1}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the storage resource ID. |
| Name | Specifies the storage controller name. |
| Description | Specifies the storage controller description information. |
| Status | Specifies the storage controller status. The options are as follows:   * **State**: indicates the storage controller enabling status. * **Health**: indicates the storage controller health status. |
| SpeedGbps | Specifies the storage controller interface rate. |
| FirmwareVersion | Specifies the storage controller firmware version. |
| SupportedDeviceProtocols | Specifies the protocol type supported by the storage controller. The options are as follows:   * SPI * PCIe * AHCI * UHCI * SAS * SATA * USB * NVMe * FC * iSCSI * FCoE * NVMeOverFabrics * SMB * NFSv3 * NFSv4 * HTTP * HTTPS * FTP * SFTP |
| Manufacturer | Specifies the storage controller manufacturer. |
| Model | Specifies the storage controller model. |
| SupportedRAIDLevels | Specifies the RAID level supported by the storage controller. The options are as follows:   * RAID0 * RAID1 * RAID5 * RAID6 * RAID10 * RAID50 * RAID60 |
| Mode | Specifies the storage driver mode. The options are as follows:   * Non-RAID * RAID |
| CachePinnedState | Specifies whether cache pinned is enabled for the storage driver. |
| SASAddress | Specifies the storage controller address. |
| ConfigurationVersion | Specifies the storage controller configuration version. |
| MemorySizeMiB | Specifies the memory capacity of the storage controller. |
| MaintainPDFailHistory | Specifies the enabling status of the fault recording function of the storage driver. |
| CopyBackState | Specifies the enabling status of the copyback function of the storage controller. |
| SmarterCopyBackState | Specifies the enabling status of the SMART error copyback function of the storage controller. |
| JBODState | Specifies the enabling status of the driver pass-through function. |
| OOBSupport | Specifies whether the controller supports out-of-band management. |
| CapacitanceName | Specifies the storage controller BBU name. |
| CapacitanceStatus | Specifies the storage controller BBU status. The options are as follows:   * **Health**: indicates the storage controller BBU health status. * **State**: indicates the storage controller BBU enabling status. |
| DriverInfo | Specifies the controller driver information. |
| DDRECCCount | Specifies the correctable memory error count of the storage controller. |
| MinStripeSizeBytes | Specifies the minimum strip size supported by the controller. |
| MaxStripeSizeBytes | Specifies the maximum strip size supported by the controller. |
| Drives | Specifies the list of drives managed by the controller. |

## Querying the Memory Information

Function

Query the memory information.

Format

**Get-iBMCMemory -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the memory information.

PS C:\> **$MemoriesArray = Get-iBMCMemory -Session $session**   
 PS C:\> **$MemoriesArray**   
   
 Host : 10.1.1.2   
 Id : mainboardDIMM000   
 CapacityMiB : 16384   
 Manufacturer : Samsung   
 OperatingSpeedMhz : 2133   
 SerialNumber : 0x177E9BFD   
 PartNumber :   
 MemoryDeviceType : DDR4   
 DataWidthBits : 72   
 RankCount : 2   
 DeviceLocator : DIMM000   
 BaseModuleType : RDIMM   
 Socket : 0   
 Controller : 0   
 Channel : 0   
 Slot : 0   
 MinVoltageMillivolt : 1200   
 Technology : Synchronous| Registered (Buffered)   
 Position : mainboard   
 Status : @{Health=OK; State=Enabled}   
   
 Host : 10.1.1.2   
 Id : mainboardDIMM001   
 CapacityMiB : 16384   
 Manufacturer : Samsung   
 OperatingSpeedMhz : 2133   
 SerialNumber : 0x177E9BFE   
 PartNumber :   
 MemoryDeviceType : DDR4   
 DataWidthBits : 72   
 RankCount : 2   
 DeviceLocator : DIMM001   
 BaseModuleType : RDIMM   
 Socket : 0   
 Controller : 0   
 Channel : 0   
 Slot : 1   
 MinVoltageMillivolt : 1200   
 Technology : Synchronous| Registered (Buffered)   
 Position : mainboard   
 Status : @{Health=OK; State=Enabled}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the memory resource ID. |
| CapacityMiB | Specifies the memory capacity. The unit is MB. |
| Manufacturer | Specifies the memory resource manufacturer. |
| OperatingSpeedMhz | Specifies the memory resource rate. |
| SerialNumber | Specifies the memory resource serial number. |
| PartNumber | Specifies the memory resource part number. |
| MemoryDeviceType | Specifies the memory resource type. |
| DataWidthBits | Specifies the data bandwidth of the memory resource. |
| RankCount | Specifies the number of ranks of the memory resource. |
| DeviceLocator | Specifies the memory resource silkscreen. |
| BaseModuleType | Specifies the basic module type of the memory resource. The options are as follows:   * RDIMM * UDIMM * SO\_DIMM * LRDIMM * Mini\_RDIMM * Mini\_UDIMM * SO\_RDIMM\_72b * SO\_UDIMM\_72b * SO\_DIMM\_16b * SO\_DIMM\_32b |
| Socket | Specifies the slot number of the CPU to which the memory resource belongs. |
| Controller | Specifies the controller ID of the memory resource. |
| Channel | Specifies the channel ID of the memory resource. |
| Slot | Specifies the slot number of the memory resource. |
| MinVoltageMillivolt | Specifies the minimum voltage of the memory resource. |
| Technology | Specifies the memory technology of the memory resource. |
| Position | Specifies the memory container of the memory resource. |
| Status | Specifies the memory resource status. |

## Querying the Power Supply Information

Function

Query the power supply information.

Format

**Get-iBMCPowerSupplies -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the power supply information.

PS C:\> **$PowerSuppliesArray = Get-iBMCPowerSupplies -Session $session**   
 PS C:\> **$PowerSuppliesArray**   
   
 Host : 10.1.1.2   
 FirmwareVersion : DC:108 PFC:107   
 LineInputVoltage : 0   
 Manufacturer : LITEON   
 MemberId : 0   
 Model : PS-2152-2H   
 Name : PS1   
 PartNumber : 02131336   
 PowerCapacityWatts : 1500   
 PowerSupplyType :   
 Redundancy : {@{@odata.id=/redfish/v1/Chassis/1/Power#/Redundancy/0}}   
 SerialNumber : 2102131336CSJ3005736   
 Status : @{State=Enabled; Health=OK}   
 ActiveStandby : Active   
 DeviceLocator : PS1   
 InputAmperage : 0   
 OutputAmperage : 0   
 OutputVoltage : 0   
 Position : chassis   
 PowerInputWatts : 0   
 PowerOutputWatts : 0   
 Protocol : PSU   
   
 Host : 10.1.1.2   
 FirmwareVersion : DC:108 PFC:107   
 LineInputVoltage : 225   
 Manufacturer : LITEON   
 MemberId : 1   
 Model : PS-2152-2H   
 Name : PS2   
 PartNumber : 02131336   
 PowerCapacityWatts : 1500   
 PowerSupplyType : AC   
 Redundancy : {@{@odata.id=/redfish/v1/Chassis/1/Power#/Redundancy/0}}   
 SerialNumber : 2102131336CSJ3001326   
 Status : @{State=Enabled; Health=OK}   
 ActiveStandby : Active   
 DeviceLocator : PS2   
 InputAmperage : 0   
 OutputAmperage : 0.234375   
 OutputVoltage : 0.046875   
 Position : chassis   
 PowerInputWatts : 204   
 PowerOutputWatts : 188   
 Protocol : PSU

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| FirmwareVersion | Specifies the firmware version of the PSU. |
| LineInputVoltage | Specifies the input voltage of the PSU. |
| Manufacturer | Specifies the PSU manufacturer. |
| MemberId | Specifies the PSU ID, which uniquely identifies the PSU in the PSU list. |
| Model | Specifies the PSU model. |
| Name | Specifies the PSU name. |
| PartNumber | Specifies the PSU part number. |
| PowerCapacityWatts | Specifies the output power of the PSU. |
| PowerSupplyType | Specifies the power supply type of the PSU. The options are as follows:   * Unknown * AC * DC * ACorDC |
| Redundancy | Specifies the list of power supply redundancy groups to which the PSU belongs. The options are as follows:   * **@odata.id**: indicates the access path of the power supply redundancy group. |
| SerialNumber | Specifies the PSU serial number. |
| Status | Specifies the PSU status. The options are as follows:   * **State**: indicates whether the PSU is enabled. * **Health**: indicates the PSU health status. |
| ActiveStandby | Specifies the active/standby mode of the PSU. The options are as follows:   * Active * Standby |
| DeviceLocator | Specifies the PSU silkscreen. |
| InputAmperage | Specifies the input current of the PSU. |
| OutputAmperage | Specifies the output current of the PSU. |
| OutputVoltage | Specifies the output voltage of the PSU. |
| Position | Specifies the PSU container. |
| PowerInputWatts | Specifies the input power of the PSU. |
| PowerOutputWatts | Specifies the output power of the PSU. |
| Protocol | Specifies the protocol used by the PSU. The options are as follows:   * PSMI * PSU * PMBUS * PMBUS OEM |

## Querying the Drive Information

Function

Query the drive information.

Format

**Get-iBMCDrives -Session** *<$session>* **-StorageId** *<StorageId>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |
| *<StorageId>* | Specifies the drive ID. This parameter is optional. |

Usage Guidelines

None

Example

# Query all drive information.

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query all drive information.

PS C:\> **$Drives = Get-iBMCDrives -Session $Session**   
 PS C:\> **$Drives**   
   
 Host : 10.1.1.2   
 Id : HDDPlaneDisk0   
 Name : Disk0   
 Model : MG04ACA400N   
 Revision : FJ3J   
 Status : @{State=Enabled; Health=OK}   
 CapacityBytes : 3999999721472   
 FailurePredicted : False   
 Protocol : SATA   
 MediaType : HDD   
 Manufacturer : TOSHIBA   
 SerialNumber : 38DGK77LF77D   
 CapableSpeedGbs : 6   
 NegotiatedSpeedGbs : 12   
 PredictedMediaLifeLeftPercent :   
 IndicatorLED : Off   
 HotspareType : None   
 StatusIndicator : OK   
 Location : {@{Info=Disk0; InfoFormat=DeviceName}}   
 DriveID : 0   
 FirmwareStatus : Online   
 HoursOfPoweredUp : 6056   
 PatrolState : DoneOrNotPatrolled   
 Position : HDDPlane   
 RebuildProgress :   
 RebuildState : DoneOrNotRebuilt   
 SASAddress : {500e004aaaaaaa00, 0000000000000000}   
 SASSmartInformation :   
 SATASmartInformation : @{AttributeRevision=; AttributeRevisionNumber=; AttributeItemList=System.Object[]}   
 SpareforLogicalDrives : {}   
 TemperatureCelsius : 33   
 Type : Disk   
   
 Host : 10.1.1.2   
 Id : HDDPlaneDisk1   
 Name : Disk1   
 Model : MG04ACA400N   
 Revision : FJ3J   
 Status : @{State=Enabled; Health=OK}   
 CapacityBytes : 3999999721472   
 FailurePredicted : False   
 Protocol : SATA   
 MediaType : HDD   
 Manufacturer : TOSHIBA   
 SerialNumber : 38DFK62PF77D   
 CapableSpeedGbs : 6   
 NegotiatedSpeedGbs : 12   
 PredictedMediaLifeLeftPercent :   
 IndicatorLED : Off   
 HotspareType : None   
 StatusIndicator : OK   
 Location : {@{Info=Disk1; InfoFormat=DeviceName}}   
 DriveID : 1   
 FirmwareStatus : UnconfiguredGood   
 HoursOfPoweredUp : 6058   
 PatrolState : DoneOrNotPatrolled   
 Position : HDDPlane   
 RebuildProgress :   
 RebuildState : DoneOrNotRebuilt   
 SASAddress : {500e004aaaaaaa01, 0000000000000000}   
 SASSmartInformation :   
 SATASmartInformation : @{AttributeRevision=; AttributeRevisionNumber=; AttributeItemList=System.Object[]}   
 SpareforLogicalDrives : {}   
 TemperatureCelsius : 33   
 Type : Disk

# Query the specified drive information.

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the specified drive information.

PS C:\> $Drives = Get-iBMCDrives -Session $Session -StorageId RAIDStorage0   
PS C:\> $Drives   
   
Host : 10.1.1.2   
Id : HDDPlaneDisk0   
Name : Disk0   
Model : MG04ACA400N   
Revision : FJ3J   
Status : @{State=Enabled; Health=OK}   
CapacityBytes : 3999999721472   
FailurePredicted : False   
Protocol : SATA   
MediaType : HDD   
Manufacturer : TOSHIBA   
SerialNumber : 38DGK77LF77D   
CapableSpeedGbs : 6   
NegotiatedSpeedGbs : 12   
PredictedMediaLifeLeftPercent :   
IndicatorLED : Off   
HotspareType : None   
StatusIndicator : OK   
Location : {@{Info=Disk0; InfoFormat=DeviceName}}   
DriveID : 0   
FirmwareStatus : Online   
HoursOfPoweredUp : 6056   
PatrolState : DoneOrNotPatrolled   
Position : HDDPlane   
RebuildProgress :   
RebuildState : DoneOrNotRebuilt   
SASAddress : {500e004aaaaaaa00, 0000000000000000}   
SASSmartInformation :   
SATASmartInformation : @{AttributeRevision=; AttributeRevisionNumber=; AttributeItemList=System.Object[]}   
SpareforLogicalDrives : {}   
TemperatureCelsius : 33   
Type : Disk

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the drive resource ID. |
| Name | Specifies the drive resource name. |
| Model | Specifies the drive model. |
| Revision | Specifies the drive version information. |
| Status | Specifies the drive status. The options are as follows:   * **State**: specifies the drive enabling status. * **Health**: specifies the drive health status. |
| CapacityBytes | Specifies the drive capacity. The unit is byte. |
| FailurePredicted | Specifies whether failure prediction is enabled on the drive. The options are as follows:   * **$true** or **1**: enabled * **$false** or **0**: disabled |
| Protocol | Specifies the protocol that the drive complies with. The options are as follows:   * SPI * PCIe * AHCI * UHCI * SAS * SATA * USB * NVMe * FC * iSCSI * FCoE * NVMeOverFabrics * SMB * NFSv3 * NFSv4 * HTTP * HTTPS * FTP * SFTP |
| MediaType | Specifies the drive media type. The options are as follows:   * HDD * SSD * SMR |
| Manufacturer | Specifies the drive manufacturer. |
| SerialNumber | Specifies the drive serial number. |
| CapableSpeedGbs | Specifies the maximum rate of the drive interface. |
| NegotiatedSpeedGbs | Specifies the negotiation rate of the drive interface. |
| PredictedMediaLifeLeftPercent | Specifies the remaining lifetime percentage of the drive. |
| IndicatorLED | Specifies the status of the location indicator of the drive. The options are as follows:   * Off * Blinking |
| HotspareType | Specifies the hot spare status of the drive. The options are as follows:   * None * Global * Dedicated |
| StatusIndicator | Specifies the indicator status of the drive. The options are as follows:   * OK * Fail * Rebuild * PredictiveFailureAnalysis * Hotspare * InAFailedArray * InACriticalArray |
| Location | Specifies the drive silkscreen. |
| DriveID | Specifies the drive ID specified when the logical drive is created. |
| FirmwareStatus | Specifies the drive status. The options are as follows:   * UnconfiguredGood * UnconfigureBad * HotSpareDrive * Offline * Failed * Online * GettingCopied * JBOD * UnconfiguredShieded * HotSpareShielded * ConfiguredShielded * Foreign * Active * Standby * Sleep * DSTInProgress * SMARTOfflineDataCollection * SCTCommand * Rebuilding |
| HoursOfPoweredUp | Specifies the duration when the drive is powered on. |
| PatrolState | Specifies the drive inspection status. The options are as follows:   * DoneOrNotPatrolled * Patrolling |
| Position | Specifies the drive container. |
| RebuildProgress | Specifies the data rebuild progress of the PCIe SSD. |
| RebuildState | Specifies the data rebuild status of the PCIe SSD. The options are as follows:   * DoneOrNotRebuilt * Rebuilding |
| SASAddress | Specifies the SAS address of the drive. |
| SASSmartInformation | Specifies the SMART information of the SAS interface. |
| SATASmartInformation | Specifies the SMART information of the SATA interface. |
| SpareforLogicalDrives | Specifies the list of logical drives to which the drive belongs when the drive is a dedicated hot spare drive. |
| TemperatureCelsius | Specifies the current temperature of the drive. |
| Type | Specifies the drive type. The options are as follows:   * Disk * SDCard * PCIe SSD Card |

## Querying the NIC Information

Function

Query all NIC information.

Format

**Get-iBMCNetworkAdapters -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query all NIC information.

PS C:\> **$NetworkAdaptersArray = Get-iBMCNetworkAdapters -Session $session**   
 PS C:\> **$NetworkAdaptersArray**   
   
 Host : 10.1.1.2   
 Id : mainboardLOM   
 Manufacturer : Intel   
 Model : X722   
 Status : @{State=Enabled; Health=OK}   
 Name : LOM   
 DriverName :   
 DriverVersion :   
 CardManufacturer : xFusion   
 CardModel : 2\*10GE+2\*GE   
 DeviceLocator : LOM   
 Position : mainboard   
 NetworkTechnology : {Ethernet}   
 RootBDF : 0000:19:03.0   
 Configuration :   
 NetworkPorts : {@{Name=1; Id=1; PhysicalPortNumber=1; LinkStatus=Down; AssociatedNetworkAddresses=System.Object[]; PortType=OpticalPort; BDF=0000:1a:0   
 0.0; FirmwarePackageVersion=; DriverVersion=; DriverName=}, @{Name=2; Id=2; PhysicalPortNumber=2; LinkStatus=Down; AssociatedNetworkAdd   
 resses=System.Object[]; PortType=OpticalPort; BDF=0000:1a:00.1; FirmwarePackageVersion=; DriverVersion=; DriverName=}, @{Name=3; Id=3;   
 PhysicalPortNumber=3; LinkStatus=Up; AssociatedNetworkAddresses=System.Object[]; PortType=ElectricalPort; BDF=0000:1a:00.2; FirmwarePac   
 kageVersion=; DriverVersion=; DriverName=}, @{Name=4; Id=4; PhysicalPortNumber=4; LinkStatus=Down; AssociatedNetworkAddresses=System.Ob   
 ject[]; PortType=ElectricalPort; BDF=0000:1a:00.3; FirmwarePackageVersion=; DriverVersion=; DriverName=}}   
   
 Host : 10.1.1.2   
 Id : mainboardMEZZ1   
 Manufacturer : Intel   
 Model : 2\*82599   
 Status : @{Health=OK; State=Enabled}   
 Name : MZ312   
 DriverName :   
 DriverVersion :   
 CardManufacturer : xFusion   
 CardModel : 4\*10G Mezzanine Card   
 DeviceLocator : mainboard   
 Position : MEZZ1   
 NetworkTechnology : {Ethernet}   
 NetworkPorts : {@{Name=1; Id=1; PhysicalPortNumber=1; LinkStatus=; AssociatedNetworkAddresses=System.Object[]; PortType=}, @{Name=2; Id=2; PhysicalPor   
 tNumber=2; LinkStatus=; AssociatedNetworkAddresses=System.Object[]; PortType=}, @{Name=3; Id=3; PhysicalPortNumber=3; LinkStatus=; Asso   
 ciatedNetworkAddresses=System.Object[]; PortType=}, @{Name=4; Id=4; PhysicalPortNumber=4; LinkStatus=; AssociatedNetworkAddresses=Syste   
 m.Object[]; PortType=}}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the NIC resource ID. |
| Manufacturer | Specifies the chip manufacturer of the NIC. |
| Model | Specifies the NIC model. |
| Status | Specifies the NIC status. The options are as follows:   * **Health**: indicates the NIC health status. * **State**: indicates the NIC enabling status. |
| Name | Specifies the external name of the NIC. |
| DriverName | Specifies the driver name of the NIC. |
| DriverVersion | Specifies the driver version of the NIC. |
| CardManufacturer | Specifies the NIC manufacturer. |
| CardModel | Specifies the NIC model. |
| DeviceLocator | Specifies the NIC silkscreen. |
| Position | Specifies the NIC container. |
| NetworkTechnology | Specifies the network protocol of the NIC. The options are as follows:   * Ethernet * FC * iSCSI * FCoE * OPA * IB |
| RootBDF | Specifies the root port BDF of the NIC. |
| Configuration | Specifies the NIC configuration information. |
| NetworkPorts | Specifies the collection of network port objects related to the NIC. |

## Querying the Fan Information

Function

Query all fan information.

Format

**Get-iBMCFans -Session** *<$session>*

Parameters

| Parameter | Description |
| --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query all fan information.

PS C:\> **$FansArray = Get-iBMCFans -Session $session**   
 PS C:\> **$FansArray**   
   
 Host : 10.1.1.2   
 MemberId : 0   
 Name : Fan Module1 Front   
 Reading : 4920   
 LowerThresholdNonCritical :   
 LowerThresholdCritical :   
 LowerThresholdFatal :   
 UpperThresholdNonCritical :   
 UpperThresholdCritical :   
 UpperThresholdFatal :   
 MinReadingRange :   
 MaxReadingRange :   
 Status : @{State=Enabled; Health=OK}   
 ReadingUnits : RPM   
 PartNumber : 02311VSF   
 Position : chassis   
 SpeedRatio : 32   
   
 Host : 10.1.1.2   
 MemberId : 1   
 Name : Fan Module2 Front   
 Reading : 4800   
 LowerThresholdNonCritical :   
 LowerThresholdCritical :   
 LowerThresholdFatal :   
 UpperThresholdNonCritical :   
 UpperThresholdCritical :   
 UpperThresholdFatal :   
 MinReadingRange :   
 MaxReadingRange :   
 Status : @{State=Enabled; Health=OK}   
 ReadingUnits : RPM   
 PartNumber : 02311VSF   
 Position : chassis   
 SpeedRatio : 32

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| MemberId | Specifies the fan sensor ID. |
| Name | Specifies the fan sensor name. |
| Reading | Specifies the current reading of the fan sensor. |
| LowerThresholdNonCritical | Specifies the lower threshold for a non-critical alarm of the fan sensor. |
| LowerThresholdCritical | Specifies the lower threshold for a critical alarm of the fan sensor. |
| LowerThresholdFatal | Specifies the lower threshold for a fatal alarm of the fan sensor. |
| UpperThresholdNonCritical | Specifies the upper threshold for a non-critical alarm of the fan sensor. |
| UpperThresholdCritical | Specifies the upper threshold for a critical alarm of the fan sensor. |
| UpperThresholdFatal | Specifies the upper threshold for a fatal alarm of the fan sensor. |
| MinReadingRange | Specifies the minimum speed reading of the fan sensor. |
| MaxReadingRange | Specifies the maximum speed reading of the fan sensor. |
| Status | Specifies the fan sensor status. The options are as follows:   * **State**: indicates whether the fan sensor is enabled. * **Health**: indicates the fan sensor health status. |
| ReadingUnits | Specifies the unit of the fan speed reading of the fan sensor. The options are as follows:   * RPM * Percent |
| PartNumber | Specifies the fan sensor part number. |
| Position | Specifies the fan sensor container. |
| SpeedRatio | Specifies the current reading of the fan sensor. The unit is percent. |

## Querying the Configuration Result Resource of the Smart Provisioning Service

Function

Query the configuration result resource of the Smart Provisioning service.

Format

**Get-iBMCSPTaskResult** **-Session** *<$session>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |

Usage Guidelines

Only V5 servers whose BIOS version is 0.39 or later support this function.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the configuration result resource of the Smart Provisioning service.

PS C:\> **$Result = Get-iBMCSPTaskResult -Session $session**   
 PS C:\> **$Result**   
   
 Host : 10.1.1.2   
 Id : 1   
 Name : SP Result   
 Status : Finished   
 OSInstall :   
 Clone :   
 Recover :   
 RaidCfg : @{Progress=Successful; Detail=System.Object[]}

1. Query details about the RAID configuration result.

PS C:\> **$result.RaidCfg | fl**   
   
 Progress : Successful   
 Detail : {@{Status=Successful; Description=import raid config Successfully.; CardModel=LSI3008; StartTime=2019-01-28 14:33:21; Operation=import; EndTime=2019-01-28 14:33:34}, @{Status=Successful; Description=Get RAID configuration Successfully; CardModel=LSI3008; StartTime=2019-01-28 14:34:30; Operation=export; EndTime=2019-01-28 14:35:27}}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Id | Specifies the configuration result resource ID. |
| Name | Specifies the configuration result resource name. |
| Status | Specifies the configuration status. The options are as follows:   * **Idle**: idle status * **Init**: being initialized * **Deploying**: being deployed * **Timeout**: deployment timed out * **Finished**: deployment completed |
| OSInstall | Specifies the basic deployment result. |
| Clone | Specifies the clone configuration result. |
| Recover | Specifies the restoration configuration result. |
| RaidCfg | Specifies the RAID configuration result. |

## Modifying the Smart Provisioning Service Resource Attributes

Function

Modify the Smart Provisioning service resource attributes.

Format

**Set-iBMCSPService** **-Session** *<$session>* **-StartEnabled** *<StartEnabled>* **-SysRestartDelaySeconds** *<SysRestartDelaySeconds>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<StartEnabled>* | Specifies whether to enable the server to start from Smart Provisioning. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<SysRestartDelaySeconds>* | Specifies the system restart time. | The value is an integer greater than 0, and the unit is second. |

Usage Guidelines

* You can set the parameters of multiple services concurrently.
* Only V5 servers whose BIOS version is 0.39 or later support this function.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Modify the Smart Provisioning service resource attributes.

PS C:\> **Set-iBMCSPService -Session $session -StartEnabled $true -SysRestartDelaySeconds 60**

## Exporting the RAID Configuration of the Smart Provisioning Service

Function

Export the RAID configuration of the Smart Provisioning service.

Format

**Export-iBMCSPRAIDSetting** **-Session** *<$session>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |

Usage Guidelines

* Only scenarios involving a single LSI SAS3008 RAID controller card are supported.
* Only V5 servers whose BIOS version is 0.39 or later support this function.
* The process of exporting the RAID configuration of the Smart Provisioning service is as follows:
  1. See the examples in this chapter to complete the task.
  2. Enable Smart Provisioning. For details, see 3.71 Modifying the Smart Provisioning Service Resource Attributes.
  3. Restart the server without affecting server services. After the restart is complete, the server automatically enters Smart Provisioning to perform the upgrade task.
  4. Query task results. For details, see 3.70 Querying the Configuration Result Resource of the Smart Provisioning Service.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Export the RAID configuration of the Smart Provisioning service.

PS C:\> **Export-iBMCSPRAIDSetting -Session $Session**

## Creating the RAID Configuration of the Smart Provisioning Service

Function

Create the RAID configuration of the Smart Provisioning service.

Format

**Add-iBMCSPRAIDVolume** **-Session** *<$session>* **-Location** *<Location>* **-DeviceName** *<DeviceName>* **-CardModel** *<CardModel>* **-VolumeName** *<VolumeName>* **-CapacityMB** *<CapacityMB>* **-BootEnabled** *<BootEnabled>* **-RAIDLevel** *<RAIDLevel>* **-Drives** *<Drives>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<Location>* | Specifies the RAID controller card location information. This parameter is mandatory. | - |
| *<DeviceName>* | Specifies the RAID controller card silkscreen name. This parameter is mandatory. | - |
| *<CardModel>* | Specifies the RAID controller card model. This parameter is optional. | * LSI3008 |
| *<VolumeName>* | Specifies the logical drive name. This parameter is optional. | - |
| *<CapacityMB>* | Specifies the preset logical drive size. This parameter is optional. | - |
| *<BootEnabled>* | Specifies whether to enable the boot drive. This parameter is mandatory. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<RAIDLevel>* | Specifies the RAID level. This parameter is mandatory. | * RAID0 * RAID1 * RAID10 * RAID1E |
| *<Drives>* | Specifies the ID list of physical member drives. This parameter is mandatory. | Use commas (,) to separate multiple IDs, for example, ,@(0,1,2). |

Usage Guidelines

* Only scenarios involving a single LSI SAS3008 RAID controller card are supported.
* Only V5 servers whose BIOS version is 0.39 or later support this function.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Create the RAID configuration of the Smart Provisioning service.

PS C:\> **$Drives = ,@(0, 1)**   
PS C:\> **$VolumeName = "Volume1"**   
PS C:\> **Add-iBMCSPRAIDVolume -Session $session -Location mainboard -DeviceName RAIDCard1 `**   
  **-VolumeName $VolumeName -CapacityMB 1048576 -BootEnabled $true `**   
  **-RAIDLevel RAID1 -Drives $Drives**

1. Export the RAID configuration of the Smart Provisioning service.

PS C:\> **Export-iBMCSPRAIDSetting -Session $Session**

1. Enable the function of starting the server from Smart Provisioning.

PS C:\> **Set-iBMCSPService -Session $session -StartEnabled $true -SysRestartDelaySeconds 60**

1. Restart the server.

PS C:\> S**et-iBMCServerPower -Session $session -ResetType ForceRestart**

1. Query the configuration result during the server restart and Smart Provisioning running processes.

PS C:\> **Get-iBMCSPTaskResult -Session $session**

1. After the Smart Provisioning running is complete, query the RAID configuration resource of the Smart Provisioning service.

PS C:\> **Get-iBMCSPRAIDSetting -Session $Session**

## Deleting the RAID Configuration of the Smart Provisioning Service

Function

Delete the RAID configuration of the Smart Provisioning service.

Format

**Clear-iBMCSPRAIDSetting** **-Session** *<$session>* **-Location** *<Location>* **-DeviceName** *<DeviceName>* **-CardModel** *<CardModel>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<Location>* | Specifies the RAID controller card location information. This parameter is mandatory. | - |
| *<DeviceName>* | Specifies the RAID controller card silkscreen name. This parameter is mandatory. | - |
| *<CardModel>* | Specifies the RAID controller card model. This parameter is optional. | * LSI3008 |

Usage Guidelines

* Only scenarios involving a single LSI SAS3008 RAID controller card are supported.
* Only V5 servers whose BIOS version is 0.39 or later support this function.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Delete the RAID configuration of the Smart Provisioning service.

PS C:\> **Clear-iBMCSPRAIDSetting -Session $session -Location mainboard -DeviceName RAIDStorage1**

1. Export the RAID configuration of the Smart Provisioning service.

PS C:\> **Export-iBMCSPRAIDSetting -Session $Session**

1. Enable the function of starting the server from Smart Provisioning.

PS C:\> **Set-iBMCSPService -Session $session -StartEnabled $true -SysRestartDelaySeconds 60**

1. Restart the server.

PS C:\> S**et-iBMCServerPower -Session $session -ResetType ForceRestart**

1. Query the configuration result during the server restart and Smart Provisioning running processes.

PS C:\> **Get-iBMCSPTaskResult -Session $session**

1. After the Smart Provisioning running is complete, query the RAID configuration resource of the Smart Provisioning service.

PS C:\> **Get-iBMCSPRAIDSetting -Session $Session**

## Modifying the RAID Configuration of the Smart Provisioning Service

Function

Modify the RAID configuration of the Smart Provisioning service.

Format

**Set-iBMCSPRAIDSetting** **-Session** *<$session>* **-Location** *<Location>* **-DeviceName** *<DeviceName>* ***-*CardModel** *<CardModel>* ***-*HotSpareDrives** *<HotSpareDrives>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<Location>* | Specifies the RAID controller card location information. This parameter is mandatory. | - |
| *<DeviceName>* | Specifies the RAID controller card silkscreen name. This parameter is mandatory. | - |
| *<CardModel>* | Specifies the RAID controller card model. This parameter is optional. | LSI3008 |
| *<HotSpareDrives>* | Specifies the global hot spare drive list. This parameter is mandatory. | Use commas (,) to separate multiple IDs, for example, ,@(0,1,2). |

Usage Guidelines

* Only scenarios involving a single LSI SAS3008 RAID controller card are supported.
* Only V5 servers whose BIOS version is 0.39 or later support this function.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Modify the RAID configuration of the Smart Provisioning service.

PS C:\> **$HotSpareDrives = ,@(0,1)**   
 PS C:\> **Set-iBMCSPRAIDSetting -Session $session -Location mainboard -DeviceName RAIDStorage1 -HotSpareDrives $HotSpareDrives**

1. Export the RAID configuration of the Smart Provisioning service.

PS C:\> **Export-iBMCSPRAIDSetting -Session $Session**

1. Enable the function of starting the server from Smart Provisioning.

PS C:\> **Set-iBMCSPService -Session $session -StartEnabled $true -SysRestartDelaySeconds 60**

1. Restart the server.

PS C:\> **Set-iBMCServerPower -Session $session -ResetType ForceRestart**

1. Query the configuration result during the server restart and Smart Provisioning running processes.

PS C:\> **Get-iBMCSPTaskResult -Session $session**

1. After the Smart Provisioning running is complete, query the RAID configuration resource of the Smart Provisioning service.

PS C:\> **Get-iBMCSPRAIDSetting -Session $Session**

## Querying the RAID Configuration Resource of the Smart Provisioning Service

Function

Query the RAID configuration resource of the Smart Provisioning service.

Format

**Get-iBMCSPRAIDSetting** **-Session** *<$session>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |

Usage Guidelines

* Only scenarios involving a single LSI SAS3008 RAID controller card are supported.
* Only V5 servers whose BIOS version is 0.39 or later support this function.
* Before querying the configuration resource, export the current RAID configuration of the Smart Provisioning service according to 3.72 Exporting the RAID Configuration of the Smart Provisioning Service**.**

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Export the RAID configuration of the Smart Provisioning service.

PS C:\> **Export-iBMCSPRAIDSetting -Session $Session**

1. Enable the function of starting the server from Smart Provisioning.

PS C:\> **Set-iBMCSPService -Session $session -StartEnabled $true -SysRestartDelaySeconds 60**

1. Restart the server.

PS C:\> S**et-iBMCServerPower -Session $session -ResetType ForceRestart**

1. Query the configuration result during the server restart and Smart Provisioning running processes.

PS C:\> **Get-iBMCSPTaskResult -Session $session**

1. After the Smart Provisioning running is complete, query the RAID configuration resource of the Smart Provisioning service.

PS C:\> **$tasks=Get-iBMCSPRAIDSetting -Session $Session**   
PS C:\> **$tasks**   
   
 Host : \*\*.\*\*.\*\*.\*\*   
 Id : mainboardRaidCard1   
 Name : SP RAID Current Configuration   
 CardModel : LSI3008   
 Location : mainboard   
 DeviceName : RAIDCard1   
 DriveGroupList : {}   
 GlobalHotSpare : {}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the ID of the current configuration resource of the RAID controller card. |
| Name | Specifies the name of the current configuration resource of the RAID controller card. |
| CardModel | Specifies the RAID controller card model. The options are as follows:   * LSI3008 |
| DeviceName | Specifies the silkscreen name of the RAID controller card. |
| GlobalHotSpare | Specifies the global hot spare drive list. |
| Location | Specifies the RAID controller card location information. |
| DriveGroupList | Specifies the drive group information. |

## Querying the OS Installation Configuration Resources of the Smart Provisioning Service

Function

Query information about the OS installation configuration resources of the Smart Provisioning service. This resource is under license control and can be used only with a license.

Format

**Get-iBMCOSDeployConfig** **-Session** *<$session>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |

Usage Guidelines

Only V5 servers whose BIOS version is 0.39 or later support this function.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the OS installation configuration resources of the Smart Provisioning service.

PS C:\> **$DeployConfig = Get-ibmcOSDeployConfig -Session $session**   
PS C:\> **$DeployConfig**   
   
Host : 10.10.1.2   
Id : 1   
Name : SP OS Install Parameter   
InstallMode : Recommended   
OSType : Win2016   
BootType : UEFIBoot   
CDKey : \*\*\*\*\*-\*\*\*\*\*-\*\*\*\*\*-\*\*\*\*\*-\*\*\*\*\*   
RootPwd : \*\*\*\*\*\*\*   
HostName : xfusion   
Autopart : False   
AutoPosition : True   
Language : en-US   
TimeZone : Eastern Standard Time   
Keyboard : 0x00000409   
CheckFirmware : False   
Partition : {@{Name=C; FileSystem=swap; Size=32}}   
Software : {@{FileName=iBMA }}   
NetCfg : {@{Device=; IPv4Addresses=System.Object[]; IPv6Addresses=System.Object[]; NameServers=System.Object[]}}   
Packages : {@{PackageName=System.Object[]; PatternName=System.Object[]}}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the ID of the OS installation configuration resource. |
| Name | Specifies the name of the OS installation configuration resource. |
| InstallMode | Specifies the installation mode. |
| OSType | Specifies the OS type. |
| BootType | Specifies the BIOS boot mode. |
| CDKey | Specifies the installation key. |
| RootPwd | Specifies the initial administrator password. |
| HostName | Specifies the host name. |
| Autopart | Specifies the automatic partition. |
| AutoPosition | Specifies whether automatic selection of the installation disk is supported. |
| Language | Specifies the language. |
| TimeZone | Specifies the time zone. |
| Keyboard | Specifies the keyboard. |
| CheckFirmware | Specifies the firmware check function. |
| Partition | Specifies the partition information. |
| Software | Specifies the software information. |
| NetCfg | Specifies the network configuration information. |
| Packages | Specifies the list of modules to be installed during the Linux system deployment. |

## Creating the OS Installation Configuration for the Smart Provisioning Service

Function

Create the OS installation configuration for the Smart Provisioning service. This resource is under license control and can be used only with a license.

Format

**Set-iBMCOSDeployConfig** **-Session** *<$session>* **-ConfigFileURI** *<ConfigFileURI>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<ConfigFileURI>* | Specifies the configuration file path of the OS to be deployed by using Smart Provisioning. | * The configuration file path must be a local path. * The configuration file must be in the JSON format. * For details about the parameters in the configuration file, see Table 3-44. |

Parameter description

| Parameter | Description | Value |
| --- | --- | --- |
| InstallMode | Specifies the installation mode. This parameter is mandatory. | The option is as follows:   * Recommended |
| OSType | Specifies the type of the OS to be installed. This parameter is mandatory. | The options are as follows:   * RHEL6U9 * RHEL7U3 * RHEL7U4 * RHEL7U5 * CentOS6U9 * CentOS7U3 * CentOS7U4 * CentOS7U5 * ESXi6.0 * ESXi6.5 * ESXi6.7 * SLES11SP4 * SLES12SP2 * SLES12SP3 * Ubuntu16.04 * Ubuntu16.04.1 * Ubuntu16.04.2 * Win2016 * Win2016 Standard Desktop * Win2016 Standard Core * Win2016 Datacenter Desktop * Win2016 Datacenter Core * Win2012\_R2 * Win2012\_R2 Standard Desktop * Win2012\_R2 Standard Core * Win2012\_R2 Datacenter Desktop * Win2012\_R2 Datacenter Core * EulerOSV2SP3 |
| BootType | Specifies the BIOS boot mode. This parameter is optional. | The options are as follows:   * UEFIBoot * LegacyBoot * SecureBoot |
| CDKey | Specifies the installation key of the Windows OS. This parameter is optional. | * For Windows, this parameter is optional and can be set to a 25-character value with every five characters connected by a hyphen (-). The value can contain uppercase letters (A to Z), lowercase letters (a to z), and digits (0 to 9). * For Linux, this parameter is left empty. * For VMware, this parameter is left empty. |
| RootPwd | Specifies the initial administrator password. This parameter is mandatory. | * For Windows, the value is a sting of at least six characters. * For SUSE, the value is a sting of at least six characters. * For CentOS, Redhat and EulerOS, the value a sting of at least six characters excluding #, $, and space. * For VMware, the value a string of at least seven characters. For the ESXi 6.7, the password must consist of letters, digits, and special characters. |
| HostName | Specifies the host name. This parameter is optional. | * The value contains a maximum of 15 characters, including uppercase letters (A to Z), lowercase letters (a to z), digits (0 to 9), and hyphens (-). * For Linux, this parameter is optional and takes effect only after the network is configured. * For Windows, this parameter is optional. * For VMware, this parameter is optional and takes effect only after the network is configured. |
| Language | Specifies the system language. This parameter is mandatory.  For details about examples, see Table 3-45. | The parameter is a character string. For details, see the installation guide of each OS.   * For Linux, this parameter is mandatory. * For Windows, this parameter is mandatory. * For VMware, this parameter is left empty. |
| TimeZone | Specifies the system time zone. This parameter is mandatory.  For details about examples, see Table 3-45. | The parameter is a character string. For details, see the installation guide of each OS.   * For Linux, this parameter is mandatory. * For Windows, this parameter is mandatory. * For VMware, this parameter is left empty. |
| Keyboard | Specifies the system keyboard layout. This parameter is mandatory.  For details about examples, see Table 3-45. | The parameter is a character string. For details, see the installation guide of each OS.   * For Linux, this parameter is mandatory. * For Windows, this parameter is mandatory. * For VMware, this parameter is left empty. |
| CheckFirmware | Specifies whether to verify firmware. This parameter is mandatory. | The options are as follows:   * **$true** or **1**: Verify the firmware. * **$false** or **0**: The firmware is not verified. |
| AutoPosition | Specifies whether automatic selection of the installation disk is supported. This parameter is mandatory. | The option is as follows:   * **$true** or **1** (The installation disk can only be automatically selected now.) |
| Autopart | Specifies whether the automatic partitioning is supported. This parameter is mandatory. | The options are as follows:   * For Linux and VMware, the value can be **$true** or **1**, indicating that automatic partitioning is supported. * For Windows, the value can be **$false** or **0**, indicating that automatic partitioning is not supported. |
| Partition | Specifies the partition information. This parameter is optional.  Format:  {  "Name": "string",  "FileSystem": " NTFS ",  "Size": "string"  } | Object. The value is a partition list.   * For Windows, the value of **Name** is any uppercase letter from C to Z.   The value of **FileSystem** is **NTFS**.  The value of **Size** is greater than 32. If it is set to **max**, the entire disk is used as the data disk.   * For Linux, the value of **Name** does not contain special characters <>|:& or spaces, such as **/**, **/home**, and **swap**.   The value of **FileSystem** is **ext4**, **ext3**, **ext2** or **xfs**.  The value of **Size** is greater than 0. The size of the root partition must be greater than 10, and the size of the swap partition must be greater than 1. If the value is set to **max**, the remaining space will be allocated.   * VMware does not support this function. |
| Software | Specifies the software list. This parameter is mandatory.  Format:  {  "FileName": "iBMA"  } | Object array. The value is the list of software to be installed. The option is as follows:   * iBMA |
| Device | Specifies the device information. This parameter is mandatory.  **Silkprint** is optional, and **Name** and **MAC** are mandatory. If **Silkprint** is entered, the values of **Name** and **MAC** will not be used.  Format:  {  "Name": "eth0",  "MAC":"\*\*:\*\*:\*\*:\*\*:\*\*:\*\*",  "Silkprint": {  "Location":"mainboard",  "DeviceName": "PCIeCard1",  "Port": "1"  }  } | Object. The value is the information of the device for which the network needs to be configured.   * **Name**: indicates the device name. * **MAC**: indicates the device MAC address. * **Silkprint**: indicates the NIC silkscreen information. * **Location**: indicates the location parameter. * **Device**: indicates the device name. * **Port**: indicates the port number. |
| IPv4Addresses | Specifies the IPv4 address information of the network port. This parameter is mandatory.  Format:  {  "Address":"Address", "SubnetMask":"SubnetMask", "AddressOrigin":"AddressOrigin", "Gateway":"Gateway"  } | Object array. The value is the information of the IPv4 to be configured.   * **Address**: indicates the IPv4 address. * **SubnetMash**: indicates the subnet mask. * **AddressOrigin**: indicates the mode for obtaining the IPv4 address, of which the value can be **Static** or **DHCP**. * **Gateway**: indicates the IPv4 gateway address. |
| IPv6Addresses | Specifies the IPv6 address information of the network port. This parameter is mandatory.  Format:  {  "Address":"Address", "PrefixLength":"PrefixLength", "AddressOrigin":"AddressOrigin" ,  "Gateway":"Gateway"  } | Object array. The value is the information of the IPv6 to be configured.   * **Address**: indicates the IPv6 address. * **PrefixLength**: indicates the prefix length of the IPv6 address. * **AddressOrigin**: indicates the mode for obtaining the IPv6 address, of which the value can be **Static** or **DHCP**. * **Gateway**: indicates the IPv6 gateway address. |
| NameServers | Specifies the DNS server address. This parameter is optional.  Format:  {  "DNS":"127.0.0.1"  } | Specifies the IP address of the DNS server. The value can be an IPv4 or IPv6 address. |
| PackageName | Specifies the package name. This parameter is optional.  Format:  {  "PackageName": ["gcc","aspell"]  } | Character string array. During the Linux deployment process, select the packages to be installed. The value can contain uppercase letters (A to Z), lowercase letters (a to z), digits (0 to 9), hyphens (-), underscores (\_), and spaces. One or more values can be specified. |
| PatternName | Specifies the pattern group name. This parameter is optional.  Format:  {  "PatternName": ["x11","base"]  } | Character string array. During the Linux deployment process, select the pattern groups to be installed. The value can contain uppercase letters (A to Z), lowercase letters (a to z), digits (0 to 9), hyphens (-), underscores (\_), and spaces. One or more values can be specified. |

Reference examples

| OSType | Language | TimeZone | Keyboard |
| --- | --- | --- | --- |
| RHEL/CentOS/EulerOS/Ubuntu | en\_US.UTF-8 | America/New\_York | us |
| SLES | en\_US | America/New\_York | english-us |
| Windows | en-US | Eastern Standard Time | 0x00000409 |

Usage Guidelines

Only V5 servers whose BIOS version is 0.39 or later support this function.

Example

# Deploy the OS.

1. Create the configuration of the system deployment.
   1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
   2. Create the configuration of the system deployment.

PS C:\> **$ConfigFileURI = 'C:\ibmc-os-deploy-config-winserver2016.json'**   
PS C:\> **Set-iBMCOSDeployConfig -Session $session -ConfigFileURI $ConfigFileURI**   
   
Host : 10.10.1.2   
Id : 1   
Name : SP OS Install Parameter   
InstallMode : Recommended   
OSType : Win2016   
BootType : UEFIBoot   
CDKey : \*\*\*\*\*-\*\*\*\*\*-\*\*\*\*\*-\*\*\*\*\*-\*\*\*\*\*   
RootPwd : \*\*\*\*\*\*\*   
HostName : xfusion   
Autopart : False   
AutoPosition : True   
Language : en-US   
TimeZone : Eastern Standard Time   
Keyboard : 0x00000409   
CheckFirmware : False   
Partition : {@{Name=C; FileSystem=swap; Size=32}}   
Software : {@{FileName=iBMA }}   
NetCfg : {@{Device=; IPv4Addresses=System.Object[]; IPv6Addresses=System.Object[]; NameServers=System.Object[]}}   
Packages : {@{PackageName=System.Object[]; PatternName=System.Object[]}}

1. Connect to the virtual media.

PS C:\> **$OSImageFileURI = 'nfs://10.10.10.3/winserver2016.iso'**   
PS C:\> **Connect-iBMCVirtualMedia $session -ImageFilePath $OSImageFileURI**

1. Enable the function of starting the server from Smart Provisioning.

PS C:\> **Set-iBMCSPService -Session $session -StartEnabled $true -SysRestartDelaySeconds 60**

1. Restart the OS.

PS C:\> **Set-iBMCServerPower -Session $session -ResetType ForceRestart**

## Restoring the Default Configuration of a Specified RAID Controller Card

Function

Restore the default configuration of a specified RAID controller card.

Format

**Restore-iBMCRAIDController** **-Session** *<$session>* **-StorageId** *<StorageId>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<StorageId>* | Specifies the RAID controller card ID. This parameter is mandatory. | Set this parameter based on the ID queried in 3.64 Querying the RAID Controller Card Information. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. estore the default configuration of a specified RAID controller card.

PS C:\> **Restore-iBMCRAIDController -Session $Session -StorageId RAIDStorage0**

## Modifying the Information of a Specified RAID Controller Card Resource

Function

Modify the information of a specified RAID controller card resource.

Format

**Set-iBMCRAIDController** **-Session** *<$session>* **-StorageId** *<StorageId>* **-CopyBackEnabled** *<CopyBackEnabled>* **-SmarterCopyBackEnabled** *<SmarterCopyBackEnabled>* **-JBODEnabled** *<JBODEnabled>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<StorageId>* | Specifies the RAID controller card ID. This parameter is mandatory. | Set this parameter based on the ID queried in 3.64 Querying the RAID Controller Card Information. |
| *<CopyBackEnabled>* | Specifies the copyback function enabling status. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<SmarterCopyBackEnabled>* | Specifies the enabling status of the SMART error copyback function. | * **$true** or **1**: enabled * **$false** or **0**: disabled   NOTE  If this parameter is set to **$true** or **1**, **CopyBackEnabled** must be set to **$true** or **1**. |
| *<JBODEnabled>* | Specifies the enabling status of the driver pass-through function. | * **$true** or **1**: enabled * **$false** or **0**: disabled |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Modify the information of a specified RAID controller card resource.

PS C:\> **Set-iBMCRAIDController -Session $session -StorageId RAIDStorage0 -CopyBackEnabled $true -SmarterCopyBackEnabled $true -JBODEnabled $true**

## Modifying the Attributes of a Specified Drive

Function

Modify the attributes of a specified drive.

Format

**Set-iBMCDrive** **-Session** *<$session>* **-DriveId** *<DriveId>* **-State** *<State>* **-LEDState** *<LEDState>* **-HotSpareType** *<HotSpareType>* **-VolumeId** *<VolumeId>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<DriveId>* | Specifies the drive resource ID. This parameter is mandatory. | Set this parameter based on the ID queried in 3.67 Querying the Drive Information. |
| *<State>* | Specifies the drive status. This parameter is optional. | The drive status can be switched between the following states:   * **Online** and **Offline** * **UnconfiguredGood** and **JBOD** * **UnconfigureBad** and **UnconfiguredGood**   NOTE  Before setting the drive status to **JBOD**, run the **Set-iBMCRAIDController** command to enable the JBOD function of the controller. For details, see 3.80 Modifying the Information of a Specified RAID Controller Card Resource. |
| *<LEDState>* | Specifies the location indicator status. This parameter is optional. | * Off * Blinking |
| *<HotSpareType>* | Specifies the hot spare status of the drive. This parameter is optional. | * None * Global * Dedicated |
| *<VolumeId>* | Specifies the ID of the associated logical drive when the drive is a dedicated hot spare disk.  NOTE  You do not need to set this parameter when **HotSpareType** is **None** or **Global**. | Set this parameter based on the ID queried in 3.82 Querying the Logical Drive Resource Information. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Modify the attributes of a specified drive.

* Change the drive status to **JBOD**.

PS C:\> **Set-iBMCDrive -Session $session -DriveId HDDPlaneDisk0 -State JBOD**

* Change the hot spare status of the logical drive whose ID is **LogicalDrive0** to **Dedicated**.

PS C:\> **Set-iBMCDrive -Session $session -DriveId HDDPlaneDisk0 -HotSpareType Dedicated -VolumeId LogicalDrive0**

## Querying the Logical Drive Resource Information

Function

Query the logical drive information about of a specified RAID controller card.

Format

**Get-iBMCVolume** **-Session** *<$session>* **-StorageId** *<StorageId>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<StorageId>* | Specifies the storage resource ID. | Set this parameter based on the ID queried in 3.64 Querying the RAID Controller Card Information. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the information of a specified logical drive resource.

PS C:\> **$Volumes = Get-iBMCVolume -Session $Session -StorageId RAIDStorage0**   
PS C:\> **$Volumes**   
   
 Host : 10.1.1.2   
 Id : LogicalDrive0   
 Name : LogicalDrive0   
 CapacityBytes : 1099511627776   
 VolumeType : Mirrored   
 OptimumIOSizeBytes : 1048576   
 Status : @{State=Enabled; Health=OK}   
 VolumeName : Volume-ps   
 RaidControllerID : 0   
 VolumeRaidLevel : RAID1   
 DefaultReadPolicy : NoReadAhead   
 DefaultWritePolicy : WriteBackWithBBU   
 DefaultCachePolicy : DirectIO   
 ConsistencyCheck : False   
 SpanNumber : 1   
 NumDrivePerSpan : 2   
 Spans : {@{SpanName=Span0; Drives=System.Object[]}}   
 CurrentReadPolicy : NoReadAhead   
 CurrentWritePolicy : WriteBackWithBBU   
 CurrentCachePolicy : DirectIO   
 AccessPolicy : ReadWrite   
 BootEnable : True   
 BGIEnable : True   
 SSDCachecadeVolume : False   
 SSDCachingEnable : False   
 AssociatedCacheCadeVolume : {}   
 DriveCachePolicy : Unchanged   
 OSDriveName :   
 InitializationMode : UnInit

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the ID of the logical drive resource. |
| Name | Specifies the name of the logical drive resource. |
| CapacityBytes | Specifies the capacity of the logical drive. |
| VolumeType | Specifies the redundancy type of the logical drive. The options are as follows:   * RawDevice * NonRedundant * Mirrored * StripedWithParity * SpannedMirrors * SpannedStripesWithParity |
| OptimumIOSizeBytes | Specifies the strip size of the logical drive. |
| Status | Specifies the logical drive status. The options are as follows:   * **Health**: indicates the health status of the logical drive. * **State**: indicates the enabling status of the logical drive. |
| VolumeName | Specifies the name of the logical drive. |
| RaidControllerID | Specifies the ID of the controller to which the logical drive belongs. |
| VolumeRaidLevel | Specifies the RAID level of the logical drive. The options are as follows:   * RAID0 * RAID1 * RAID2 * RAID3 * RAID4 * RAID5 * RAID6 * RAID10 * RAID1E * RAID20 * RAID30 * RAID40 * RAID50 * RAID60 |
| DefaultReadPolicy | Specifies the default read policy of the logical drive. The options are as follows:   * NoReadAhead * ReadAhead |
| DefaultWritePolicy | Specifies the default write policy of the logical drive. The options are as follows:   * WriteThrough * WriteBackWithBBU * WriteBack |
| DefaultCachePolicy | Specifies the default cache policy of the logical drive. The options are as follows:   * CachedIO * DirectIO |
| ConsistencyCheck | Specifies the enabling status of the consistency check function. |
| SpanNumber | Specifies the number of spans of the logical drive. |
| NumDrivePerSpan | Specifies the number of member drives of a span. |
| Spans | Specifies the subattributes. The options are as follows:   * **SpanName**: indicates the name of the span. * **Drives**: indicates the list of member drives contained in the span. |
| CurrentReadPolicy | Specifies the current read policy of the logical drive. |
| CurrentWritePolicy | Specifies the current write policy of the logical drive. |
| CurrentCachePolicy | Specifies the current cache policy of the logical drive. |
| AccessPolicy | Specifies the access policy of the logical drive. The options are as follows:   * ReadWrite * ReadOnly * Blocked * Hidden |
| BootEnable | Specifies whether the logical drive is the boot drive. |
| BGIEnable | Specifies the enabling status of background initialization. |
| SSDCachecadeVolume | Specifies whether the logical drive is a CacheCade logical drive. |
| SSDCachingEnable | Specifies whether a CacheCade logical drive can be used. |
| AssociatedCacheCadeVolume | Specifies the access path of the associated CacheCade logical drive. |
| DriveCachePolicy | Specifies the cache policy of the physical drive. The options are as follows:   * Enabled * Disabled * Unchanged |
| OSDriveName | Specifies the OS drive letter corresponding to the logical drive. |
| InitializationMode | Specifies the initialization mode of the logical drive. The options are as follows:   * UnInit * QuickInit * FullInit |

## Modifying the Logical Drive Resource Attributes

Function

Modify the logical drive resource attributes of a specified RAID controller card.

Format

**Set-iBMCVolume** **-Session** *<$session>* **-StorageId** *<StorageId>* **-VolumeId** *<VolumeId>* **-VolumeName** *<VolumeName>* **-DefaultReadPolicy** *<DefaultReadPolicy>* **-DefaultWritePolicy** *<DefaultWritePolicy>* **-DefaultCachePolicy** *<DefaultCachePolicy>* **-AccessPolicy** *<AccessPolicy>* **-DriveCachePolicy** *<DriveCachePolicy>* **-BootEnabled** *<BootEnabled>* **-BGIEnabled** *<BGIEnabled>* **-SSDCachingEnabled** *<SSDCachingEnabled>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<StorageId>* | Specifies the storage resource ID. | Set this parameter based on the ID queried in 3.64 Querying the RAID Controller Card Information. |
| *<VolumeId>* | Specifies the logical drive resource ID. This parameter is mandatory. | Set this parameter based on the ID queried in 3.82 Querying the Logical Drive Resource Information. |
| *<VolumeName>* | Specifies the logical drive name. This parameter is optional. | - |
| *<DefaultReadPolicy>* | Specifies the default read policy of the logical drive. This parameter is optional. | * **NoReadAhead**: The Read Ahead function is disabled. * **ReadAhead**: The Read Ahead function is enabled. The controller pre-reads sequential data or the data predicted to be used and saves it in the cache. |
| *<DefaultWritePolicy>* | Specifies the default write policy of the logical drive. This parameter is optional. | * **WriteThrough**: After the drive subsystem receives all data, the controller sends the host a signal indicating that data transmission is complete. * **WriteBackWithBBU**: When no BBU is configured for the controller or the BBU is faulty, the controller automatically switches to the **WriteThrough** mode. * **WriteBack**: After receiving all the transmitted data, the controller cache returns a data transmission completion signal to the host. |
| *<DefaultCachePolicy>* | Specifies the default cache policy of the logical drive. This parameter is optional. | * **CachedIO**: All the read and write requests are processed by the RAID controller card cache. Use this value only when CacheCade 1.1 is configured. This value is not recommended in other scenarios. * **DirectIO**: This value has different definitions in read and write scenarios.   + - In read scenarios, data is directly read from physical drives. When **DefaultReadPolicy** is set to **ReadAhead**, data read requests are processed by the RAID controller card cache.     - In write scenarios, data write requests are processed by the RAID controller card cache. When **DefaultWritePolicy** is set to **WriteThrough**, data write requests are not processed by the RAID controller card cache and data is directly written into physical drives. |
| *<AccessPolicy>* | Specifies the access policy of the logical drive. This parameter is optional. | * **ReadWrite**: The logical drive is readable and writable. * **ReadOnly**: The logical drive can only be read. * **Blocked**: Access to the logical drive is denied. |
| *<DriveCachePolicy>* | Specifies the cache policy of the member drive. This parameter is optional. | * **Unchanged**: The default cache policy is retained. * **Enabled**: During the read/write process, data is written to the cache before being written to physical drives, improving the write performance. However, if the system is powered off unexpectedly, data will be lost if no protection mechanism is available. * **Disabled**: During the read/write process, data is not written to the cache before being written to physical drives. When the system is powered off unexpectedly, data will not be lost. |
| *<BootEnabled>* | Specifies whether the drive is the boot drive. This parameter is optional. | This parameter can be set to **$true** or **1**, indicating that the drive is the boot drive. |
| *<BGIEnabled>* | Specifies whether to enable background initialization. This parameter is optional. | * **$true** or **1**: enable * **$false** or **0**: disable |
| *<SSDCachingEnabled>* | Specifies whether to use the CacheCade logical drive as the cache. This parameter is optional. | * **$true** or **1**: The CacheCade logical drive is used as the cache. * **$false** or **0**: The CacheCade logical drive is not used as the cache. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Modify the logical drive resource attributes of a specified RAID controller card.

PS C:\> **Set-iBMCVolume -Session $session -StorageId RAIDStorage0 -VolumeId LogicalDrive0 -VolumeName Volume1 `**   
 **-DefaultCachePolicy CachedIO -DefaultWritePolicy WriteBack -DefaultReadPolicy ReadAhead `**   
 **-AccessPolicy ReadOnly -DriveCachePolicy Enabled `**   
 **-BootEnabled $true -BGIEnabled $true**

## Initializing a Logical Drive

Function

Initialize the logical drive of a specified RAID controller card.

Format

**Initialize-iBMCVolume** **-Session** *<$session>* **-StorageId** *<StorageId>* **-VolumeId** *<VolumeId>* **-InitAction** *<InitAction>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<StorageId>* | Specifies the storage resource ID. This parameter is mandatory. | Set this parameter based on the ID queried in 3.64 Querying the RAID Controller Card Information. |
| *<VolumeId>* | Specifies the logical drive ID. This parameter is mandatory. | Set this parameter based on the ID queried in 3.82 Querying the Logical Drive Resource Information. |
| *<InitAction>* | Specifies the initialization mode. This parameter is mandatory. | The following initialization modes are supported:   * **QuickInit**: indicates the quick initialization. No task will be created. * **FullInit**: indicates the complete initialization. A task will be created. * **CancelInit**: indicates that the initialization is canceled. No task will be created. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Initialize the logical drive of a specified RAID controller card.

PS C:\> **Initialize-iBMCVolume -Session $session -StorageId RAIDStorage0 `**   
 **-VolumeId LogicalDrive0 -InitAction QuickInit**

## Deleting a Logical Drive

Function

Delete the logical drive of a specified RAID controller card.

Format

**Remove-iBMCVolume** **-Session** *<$session>* **-StorageId** *<StorageId>* **-VolumeId** *<VolumeId>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<StorageId>* | Specifies the storage resource ID. | Set this parameter based on the ID queried in 3.64 Querying the RAID Controller Card Information. |
| *<VolumeId>* | Specifies the logical drive resource ID. This parameter is mandatory. | Set this parameter based on the ID queried in 3.82 Querying the Logical Drive Resource Information. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Delete the logical drive of a specified RAID controller card.

PS C:\> **Remove-iBMCVolume -Session $session -StorageId RAIDStorage0 -VolumeId LogicalDrive0**   
   
Host : 10.1.1.2   
Id : 3   
Name : volume deletion task   
ActivityName : [10.1.1.2] volume deletion task   
TaskState : Completed   
StartTime : 2019-01-06T22:48:09+00:00   
EndTime : 2019-01-06T22:48:13+00:00   
TaskStatus : OK   
TaskPercent :

## Creating a Logical Drive

Function

Create a logical drive under a specified RAID controller card.

Format

**Add-iBMCVolume** **-Session** *<$session>* **-StorageId** *<StorageId>* **-CacheCade** *<CacheCade>* **-RAIDLevel** *<RAIDLevel>* **-Drives** *<Drives>* **-VolumeName** *<VolumeName>* **-StripSize** *<StripSize>* **-CapacityMB** *<CapacityMB>* **-DefaultReadPolicy** *<DefaultReadPolicy>* **-DefaultWritePolicy** *<DefaultWritePolicy>* **-DefaultCachePolicy** *<DefaultCachePolicy>* **-AccessPolicy** *<AccessPolicy>* ***-DriveCachePolicy****<DriveCachePolicy>* **-InitMode** *<InitMode>* ***-SpanNumber*** *<SpanNumber>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<StorageId>* | Specifies the storage resource ID. This parameter is mandatory. | Set this parameter based on the ID queried in 3.64 Querying the RAID Controller Card Information. |
| *<CacheCade>* | Specifies whether the logical drive is created as a CacheCade logical drive. | * When a CacheCade logical drive is to be created, set this parameter to **$true** or **1**. * When a non-CacheCade logical drive is to be created, this parameter is optional and can be set to **$false** or **0**. |
| *<RAIDLevel>* | Specifies the RAID level of the logical drive. This parameter is mandatory. | * RAID0 * RAID1 * RAID5 * RAID6 * RAID10 * RAID50 * RAID60   NOTE   * When a CacheCade logical drive is to be created, this parameter can only be set to **RAID0** or **RAID1**. * You do not need to set this parameter when adding a logical drive to an existing drive group. |
| *<Drives>* | Specifies the list of member drives. This parameter is mandatory. | Format: [disk1,disk2,...,diskN]  NOTE   * The member drives must use the same type of interfaces and storage media. * Enter the ID of a physical drive in the available drive group if you want to add a logical drive to an existing drive group. * Set this parameter based on the **DriveID** queried in 3.67 Querying the Drive Information, when **FirmwareStatus** is set to **UnconfiguredGood**. |
| *<VolumeName>* | Specifies the logical drive name. This parameter is optional. | - |
| *<StripSize>* | Specifies the strip size of the logical drive. This parameter is optional. | The unit is byte. The options are as follows:   * 65536 * 131072 * 262144 * 524288 * 1048576   NOTE  You do not need to set this parameter when creating a CacheCade logical drive. |
| *<CapacityMB>* | Specifies the logical drive capacity. This parameter is optional. | The value must be an integer in bytes. The value must be a multiple of 1048576 (1 MB).  NOTE  You do not need to set this parameter when creating a CacheCade logical drive. |
| *<DefaultReadPolicy>* | Specifies the default read policy of the logical drive. This parameter is optional. | * **NoReadAhead**: The Read Ahead function is disabled. * **ReadAhead**: The Read Ahead function is enabled. The controller pre-reads sequential data or the data predicted to be used and saves it in the cache.   NOTE  You do not need to set this parameter when creating a CacheCade logical drive. |
| *<DefaultWritePolicy>* | Specifies the default write policy of the logical drive. This parameter is optional. | * **WriteThrough**: After the drive subsystem receives all data, the controller sends the host a signal indicating that data transmission is complete. * **WriteBackWithBBU**: When no BBU is configured for the controller or the BBU is faulty, the controller automatically switches to the **WriteThrough** mode. * **WriteBack**: After receiving all the transmitted data, the controller cache returns a data transmission completion signal to the host.   NOTE  You cannot set this parameter to **WriteBack** when creating a CacheCade logical drive. |
| *<DefaultCachePolicy>* | Specifies the default cache policy of the logical drive. This parameter is optional. | * **CachedIO**: All the read and write requests are processed by the RAID controller card cache. Use this value only when CacheCade 1.1 is configured. This value is not recommended in other scenarios. * **DirectIO**: This value has different definitions in read and write scenarios.   + - In read scenarios, data is directly read from physical drives. When **DefaultReadPolicy** is set to **ReadAhead**, data read requests are processed by the RAID controller card cache.     - In write scenarios, data write requests are processed by the RAID controller card cache. When **DefaultWritePolicy** is set to **WriteThrough**, data write requests are not processed by the RAID controller card cache and data is directly written into physical drives. |
| *<AccessPolicy>* | Specifies the access policy of the logical drive. This parameter is optional. | The following policies are supported:   * **ReadWrite**: The logical drive is readable and writable. * **ReadOnly**: The logical drive can only be read. * **Blocked**: Access to the logical drive is denied.   NOTE  You do not need to set this parameter when creating a CacheCade logical drive. |
| *<DriveCachePolicy>* | Specifies the cache policy of the member drive. This parameter is optional. | The following policies are supported:   * **Unchanged**: The default cache policy is retained. * **Enabled**: During the read/write process, data is written to the cache before being written to physical drives, improving the write performance. However, if the system is powered off unexpectedly, data will be lost if no protection mechanism is available. * **Disabled**: During the read/write process, data is not written to the cache before being written to physical drives. When the system is powered off unexpectedly, data will not be lost.   NOTE  You do not need to set this parameter when creating a CacheCade logical drive. |
| *<InitMode>* | Specifies the initialization mode of the logical drive. This parameter is optional. | * **UnInit**: The initialization is not performed. * **QuickInit**: Write zeros for the first and last 10 MB of the logical drive. After that, the logical drive status changes to **Optimal**. * **FullInit**: Initialize the logical drive. Before the initialization is complete, the logical drive status is **initialization**.   NOTE  You do not need to set this parameter when creating a CacheCade logical drive. |
| *<SpanNumber>* | Specifies the number of spans of the logical drive. This parameter is optional. | * Set this parameter to **1** if the RAID level is RAID 0, RAID 1, RAID 5, or RAID 6. * Set this parameter to a value ranging from **2** to **8** if the RAID level is RAID 10, RAID 50, or RAID 60.   NOTE   * You do not need to set this parameter when creating a CacheCade logical drive. * You do not need to set this parameter when adding a logical drive to an existing drive group. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Create a logical drive under a specified RAID controller card.

PS C:\> **$Drives = ,@(0, 1)**   
PS C:\> **$Volumes = Add-iBMCVolume -Session $session -StorageId RAIDStorage0 -CacheCade $false -RAIDLevel RAID1 -Drives $Drives**   
   
 Host : 10.1.1.2   
 Id : 4   
 Name : volume creation task   
 ActivityName : [10.1.1.2] volume creation task   
 TaskState : Completed   
 StartTime : 2019-01-07T09:37:15+00:00   
 EndTime : 2019-01-07T09:37:29+00:00   
 TaskStatus : OK   
 TaskPercent :

## Collecting iBMC Logs in One-click Mode

Function

Collect the maintenance information about all modules of a board.

Format

**Export-iBMCMaintenanceInfo -Session** *<$session>* **-ExportTo** *<ExportTo>* **-SecureEnabled**

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<ExportTo>* | Specifies the path for exporting the configuration file. This parameter is mandatory. | * If a local temporary directory of the iBMC is used, the directory must be the **/tmp** directory and a file name must be specified. * Using a remote directory:   + - When **-SecureEnabled** is configured, the dialog box for entering the user name and password of the remote protocol is displayed. The format of the file path is *File transfer protocol://Remote server IP address/Directory/File name*.     - When **-SecureEnabled** is not configured, the format of the file path is *File transfer protocol://User name:Password@Remote server IP address/Directory/File name*.     - File transfer protocols include HTTPS, SCP, SFTP, CIFS, and NFS. |
| -SecureEnabled | (Optional) Whether to enable the interactive function. The function is enabled when this parameter is added to the command. You are advised to enable this function when using a remote directory to ensure operation security. | - |

Usage Guidelines

The format of the exported file must be .tar.gz.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Export iBMC logs.

* Export iBMC logs to the **tmp** directory of the iBMC.

PS C:\> **$ExportPath = "/tmp/collect.tar.gz"**   
 PS C:\> **$Tasks = Export-iBMCMaintenanceInfo -Session $session -ExportTo $ExportPath**   
 PS C:\> **$Tasks**   
   
 Host : 10.1.1.2   
 Id : 1   
 Name : Export Dump File Task   
 ActivityName : [10.1.1.2] Export Dump File Task   
 TaskState : Completed   
 StartTime : 2019-01-19T04:22:13+00:00   
 EndTime : 2019-01-19T04:30:19+00:00   
 TaskStatus : OK   
 TaskPercent : 100%

* Export iBMC logs a remote directory.

PS C:\> **$ExportPath = "nfs://10.10.10.3/data/nfs/collect.tar.gz"**   
 PS C:\> **$Tasks = Export-iBMCMaintenanceInfo -Session $session -ExportTo $ExportPath**   
 PS C:\> **$Tasks**   
   
 Host : 10.1.1.2   
 Id : 1   
 Name : Export Dump File Task   
 ActivityName : [10.1.1.2] Export Dump File Task   
 TaskState : Completed   
 StartTime : 2019-01-19T04:22:13+00:00   
 EndTime : 2019-01-19T04:30:19+00:00   
 TaskStatus : OK   
 TaskPercent : 100%

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the task ID. |
| Name | Specifies the task name. |
| ActivityName | Specifies the task connection status. |
| TaskState | Specifies the task execution status. The options are as follows:   * New * Starting * Running * Suspended * Interrupted * Pending * Stopping * Completed * Killed * Exception * Service |
| StartTime | Specifies the task start time. |
| EndTime | Specifies the task end time. |
| TaskStatus | Specifies the task execution status. |
| TaskPercent | Specifies the task execution progress. |

## Downloading a BMC File

Function

Download a BMC file.

Format

**Invoke-iBMCFileDownload -Session** *<$session>* **-BMCFileUri** *<BMCFileUri>* **-LocalFileUri** *<LocalFileUri>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<BMCFileUri>* | Specifies the path of the file in the BMC. | The value is **/tmp/web** and its subdirectories. |
| *<LocalFileUri>* | Specifies the path for storing the downloaded file. | The value is a local directory. The value is in the format of *Directory\File name* or \\*Server IP address*\*Directory*\*File name*. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Download a BMC file.

PS C:\> **$BMCFilePath = "/tmp/web/2288H\_V5\_5288\_V5-iBMC-V318.hpm"**   
 PS C:\> **$LocalFilePath = "E:\2288H\_V5\_5288\_V5-iBMC-V318.hpm"**   
 PS C:\> **Invoke-iBMCFileDownload -Session $session -BMCFileUri $BMCFilePath -LocalFileUri $LocalFilePath**

## Uploading a File

Function

Use a Redfish interface to upload a file, and store the file in the **/tmp/web** directory.

Format

**Invoke-iBMCFileUpload -Session** *<$session>* **-FileUri** *<FileUri>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<FileUri>* | Specifies the path of the file. | Select the corresponding file in the local directory.   * The file types supported by a x86 V3 board are as follows:   .hpm, .cer, .pem, .cert, .crt, .pfx, .p12, .xml, .keys, and .pub   * The file types supported by a x86 V5 board are as follows:   .hpm, .zip, .asc, .cer, .pem, .cert, .crt, .pfx, .p12, .xml, .keys, and .pub   * The maximum size of an. hpm file supported by a x86 V3 board is 46 MB. The maximum size of an .hpm, .zip, or .asc file supported by a V5 board is 60 MB. * The maximum size of a .cer, .pem, .cert, .crt, .xml, or .p12 file is 1 MB. The maximum size of a .pfx or .keys file is 2 MB. The maximum size of a .pub file is 2 KB. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Upload a file to the **/tmp/web** directory.

PS C:\> **$Path = Invoke-iBMCFileUpload -Session $session -FileUri E:\2288H\_V5\_5288\_V5-iBMC-V318.hpm**   
 PS C:\> **$Path**   
   
 Path   
 ----   
 /tmp/web/2288H\_V5\_5288\_V5-iBMC-V318.hpm

## Querying the License Service Information

Function

Query the license service information.

Format

**Get-iBMCLicense** **-Session** *<$session>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |

Usage Guidelines

V3 servers and some V5 servers do not support this function. For details, see the [iBMC User Guide](https://support.xfusion.com/support/#/en/server-management-software/ibmc-pid-8060757/document?category=C12&subcategory=C0025).

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the license service information.

PS C:\> **$License = Get-iBMCLicense -Session $session**   
PS C:\> **$License**   
   
Host : 10.1.1.2   
Id : LicenseService   
Name : License Service   
Capability : {Local, Remote}   
DeviceESN : A5A7D970F13580158FBBCA5009F17234D1A6C5CB   
InstalledStatus : Installed   
RevokeTicket :   
LicenseClass : Advanced   
LicenseStatus : Normal   
LicenseInfo : @{FileFormatVersion=3.10; GeneralInfo=; CustomerInfo=; NodeInfo=; SaleInfo=}   
AlarmInfo : @{RemainGraceDay=0; RemainCommissioningDay=0; ProductESNValidState=Valid; FileState=Normal; ProductESNMatchState=Matched; ProductVersionMatchState=Matched}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the host name. |
| Id | Specifies the ID of a license service resource. |
| Name | Specifies the name of a license service resource. |
| Capability | Specifies the license management capabilities. The options are as follows:   * **Local**: supports the local license management. * **Remote**: supports the remote license management. |
| DeviceESN | Specifies the ESN of a device. |
| InstalledStatus | Specifies the installation status of a license file. The options are as follows:   * NotInstalled * Installed * Installing |
| RevokeTicket | Specifies the license revocation code. |
| LicenseClass | Specifies the license file level when the license file is managed locally. When it is managed remotely, this field is invalid. The options are as follows:   * NotInstalled * Advanced |
| LicenseStatus | Specifies the license status. The options are as follows:   * N/A * Default * GracePeriod * Normal * Emergency * Commissioning * Unknown |
| LicenseInfo | Specifies the license file information. |
| AlarmInfo | Specifies the license alarm information. |

## Installing a License

Function

Install a license.

Format

**Install-iBMCLicense** **-Session** *<$session>* **-LicenseFileURI** *<LicenseFileURI>* **-LicenseSource** *<LicenseSource>* **-SecureEnabled**

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<LicenseFileURI>* | Specifies the path of the license file. | * If a local directory is used, the value must be in *Directory\File name* or *\\Server IP address\Directory\File name* format. * If a local temporary directory of the iBMC is used, the directory must be the **/tmp** directory and a file name must be specified. * Using a remote directory:   + - When **-SecureEnabled** is configured, the dialog box for entering the user name and password of the remote protocol is displayed. The format of the file path is *File transfer protocol://Remote server IP address/Directory/File name*.     - When **-SecureEnabled** is not configured, the format of the file path is *File transfer protocol://User name:Password@Remote server IP address/Directory/File name*.     - The file transfer protocols include HTTPS, SCP, SFTP, CIFS, and NFS, and the protocol name must be in lower case. |
| *<LicenseSource>* | Specifies the source of the license file. | The options are as follows:   * iBMC * FusionDirector * eSight   If the parameter is not configured, the default value is **iBMC**. |
| -SecureEnabled | (Optional) Whether to enable the interactive function. The function is enabled when this parameter is added to the command. You are advised to enable this function when using a remote directory to ensure operation security. | - |

Usage Guidelines

V3 servers and some V5 servers do not support this function. For details, see the [iBMC User Guide](https://support.xfusion.com/support/#/en/server-management-software/ibmc-pid-8060757/document?category=C12&subcategory=C0025).

Example

# Use a local file to install a license.

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Install a license.

PS C:\> **Install-iBMCLicense -Session $session `**   
 **-LicenseFileURI "E:\xfusion\PowerShell\LIC2288H\_V5\_2\_20180905LTM65C.xml" `**   
 **-LicenseSource iBMC**

# Use the iBMC local temporary directory to install the license.

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Install a license.

PS C:\> **Install-iBMCLicense -Session $session `**   
 **-LicenseFileURI "/tmp/LIC2288H\_V5\_2\_20180905LTM65C.xml"**

# Use the remote directory to install a license.

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Install a license.

PS C:\> **Install-iBMCLicense -Session $session `**   
 **-LicenseFileURI "nfs://10.10.10.2/data/nfs/LIC2288H\_V5\_2\_20180905LTM65C.xml"**

## Exporting a License

Function

Export a license.

Format

**Export-iBMCLicense -Session** *<$session>* **-ExportTo** *<ExportTo>* **-SecureEnabled**

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<ExportTo>* | Specifies the path of the exported file. | * If a local temporary directory of the iBMC is used, the directory must be the **/tmp** directory and a file name must be specified. * Using a remote directory:   + - When **-SecureEnabled** is configured, the dialog box for entering the user name and password of the remote protocol is displayed. The format of the file path is *File transfer protocol://Remote server IP address/Directory/File name*.     - When **-SecureEnabled** is not configured, the format of the file path is *File transfer protocol://User name:Password@Remote server IP address/Directory/File name*.     - File transfer protocols include HTTPS, SCP, SFTP, CIFS, and NFS. |
| -SecureEnabled | (Optional) Whether to enable the interactive function. The function is enabled when this parameter is added to the command. You are advised to enable this function when using a remote directory to ensure operation security. | - |

Usage Guidelines

V3 servers and some V5 servers do not support this function. For details, see the [iBMC User Guide](https://support.xfusion.com/support/#/en/server-management-software/ibmc-pid-8060757/document?category=C12&subcategory=C0025).

Example

# Use the iBMC local temporary directory to export a license.

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Export a license.

PS C:\> **Export-iBMCLicense -Session $session -ExportTo "/tmp/License.xml"**

# Use the remote directory to export a license.

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Export a license.

PS C:\> **Export-iBMCLicense -Session $session -ExportTo "NFS://192.168.2.10/data/nfs/License.xml"**

## Deleting a License

Function

Delete a license.

Format

**Remove-iBMCLicense** **-Session** *<$session>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |

Usage Guidelines

V3 servers and some V5 servers do not support this function. For details, see the [iBMC User Guide](https://support.xfusion.com/support/#/en/server-management-software/ibmc-pid-8060757/document?category=C12&subcategory=C0025).

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Delete a license.

PS C:\> **Remove-iBMCLicense -Session $session**

## Querying iBMC Network Port Information

Function

This command is used to query iBMC network port information. Currently, only iBMC management network port information can be queried.

Format

**Get-iBMCIP** **-Session** *<$session>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | (Mandatory) Session | - |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query information about the iBMC network port.

PS C:\Users\Administrator> **Get-iBMCIP -Session $session**   
   
   
Host : 172.26.100.8   
Id : 04338932ede2   
Name : Manager Ethernet Interface   
PermanentMACAddress : \*\*:\*\*:\*\*:\*\*:\*\*:\*\*   
HostName : server8   
FQDN : server8.plugin.com   
VLAN : @{VLANEnable=False; VLANId=0}   
IPv4Addresses : {@{Address=172.26.100.8; SubnetMask=255.255.0.0; Gateway=172.26.0.1; AddressOrigin=Static}}   
IPv6Addresses : {@{Address=\*\*\*:\*\*::\*; PrefixLength=64; AddressOrigin=Static}, @{Address=fe80::633:89ff:fe32:ede2; PrefixLength=64; AddressOrigin=LinkLocal}}   
IPv6StaticAddresses : {@{Address=\*\*\*:\*\*::\*; PrefixLength=64}}   
IPv6DefaultGateway : \*\*\*:\*\*::\*   
NameServers : {172.26.201.7, }   
IPVersion : IPv4AndIPv6   
NetworkPortMode : Fixed   
ManagementNetworkPort : @{Type=Dedicated; PortNumber=1}

Output Description

Output description

| Output | Description |
| --- | --- |
| Host | Host name |
| Id | Task ID |
| Name | Task name |
| PermanentMACAddress | MAC address of the iBMC network port |
| HostName | Host name of the iBMC |
| FQDN | Fully qualified domain name (FQDN) of the iBMC |
| VLAN | VLAN information of the iBMC network port   * **VLANEnable**: indicates whether VLAN is enabled. * **VLANId**: VLAN ID. |
| IPv4Addresses | IPv4 address information of the iBMC network port   * **Address**: IPv4 address * **SubnetMask**: subnet mask of the IPv4 address * **Gateway**: IPv4 gateway address * **AddressOrigin**: acquisition mode of the IPv4 address |
| IPv6Addresses | IPv6 address information of the iBMC network port   * **Address**: IPv6 address * **PrefixLength**: prefix length of the IPv6 address * **AddressOrigin**: acquisition mode of the IPv6 address |
| IPv6StaticAddresses | Static IPv6 address information of the iBMC network port   * **Address**: IPv6 address * **PrefixLength**: prefix length of the IPv6 address |
| IPv6DefaultGateway | IPv6 gateway address of the iBMC network port |
| NameServers | Addresses of the preferred and alternate DNS servers when the iBMC network port address is dynamically assigned by using the DHCP. The server IP address can be an IPv4 or IPv6 address. |
| IPVersion | IP versions to be used |
| NetworkPortMode | Network port mode. Value:   * **Fixed** * **Automatic** |
| ManagementNetworkPort | Management network Port   * **Type**: type of the management network port * **PortNumber**: silkscreen of the management network port   NOTE  If a dedicated network port is used, **PortNumber** indicates the serial number of the port, not the silkscreen. |

## Setting iBMC Network port Information

Function

This command is used to set iBMC network port information. Currently, only iBMC management network port information can be set.

Format

**Set-iBMCIP -Session** *<$session>* **-IPVersion** *<IPVersion>* **-IPv4Address** *<IPv4Address>* **-IPv4SubnetMask** *<IPv4SubnetMask>* **-IPv4Gateway** *<IPv4Gateway>* **-IPv4AddressOrigin***<IPv4AddressOrigin>* **-IPv6Address** *<IPv6Address>* **-IPv6PrefixLength** *<IPv6PrefixLength>* **-IPv6Gateway***<IPv6Gateway>* **-IPv6AddressOrigin**  *<IPv6AddressOrigin>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | (Mandatory) Session | - |
| *<IPVersion>* | (Optional) IP versions to be used | * **IPv4**: enable IPv4 only. * **IPv6**: enable IPv6 only. * **IPv4AndIPv6**: enable IPv4 and IPv6. |
| *<IPv4Address>* | (Optional) IPv4 address of the iBMC network port | - |
| *<IPv4SubnetMask>* | (Optional) Subnet mask of the IPv4 address | - |
| *<IPv4Gateway>* | (Optional) IPv4 gateway address of the iBMC network port | - |
| *<IPv4AddressOrigin>* | (Optional) Mode in which the IPv4 address of the iBMC network port is obtained. | * Static * DHCP |
| *<IPv6Address>* | (Optional) IPv6 address of the iBMC network port | - |
| *<IPv6PrefixLength>* | (Optional) Prefix length of the IPv6 address | - |
| *<IPv6Gateway>* | (Optional) IPv6 gateway address of the iBMC network port | - |
| *<IPv6AddressOrigin>* | (Optional) Mode in which the IPv6 address of the iBMC network port is obtained. | * Static * DHCPv6 |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Configure the iBMC network port information.

PS C:\> **Set-iBMCIP -Session $Session -IPVersion IPv4AndIPv6 -IPv4Address 10.1.1.2 -IPv4SubnetMask 255.255.0.0 -IPv4Gateway 10.1.0.1 -IPv4AddressOrigin Static -IPv6Address fc00::10:2 -IPv6PrefixLength 64 -IPv6Gateway fc00::10:1 -IPv6AddressOrigin Static**

## Querying LDAP Information

Function

Query LDAP information.

Format

**Get-iBMCLDAP** **-Session** *<$session>* **-LDAPID** *<LDAPID>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<LDAPID>* | Specifies the ID of the LDAP resource. This parameter is optional.  NOTE  If no LDAP ID is specified, information about all domain controllers will be queried. | An integer ranging from 1 to 6. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query LDAP information.

PS C:\> **$LDAPID = 1**   
PS C:\> **Get-iBMCLDAP -Session $Session -LDAPID $LDAPID**   
   
Host : 10.1.1.2   
Id : 1   
Name : Ldap Controller   
LdapServerAddress :   
LdapPort : 636   
UserDomain : ,DC=   
BindDN :   
BindPassword :   
CertificateVerificationEnabled : False   
CertificateVerificationLevel :   
CertificateInformation :   
CertificateChainInformation :   
LdapGroups : {@{MemberId=0; GroupName=; GroupDomain=CN=,OU=,DC=; GroupRole=No Access;   
GroupLoginRule=System.Object[];   
 GroupLoginInterface=System.Object[]}, @{MemberId=1; GroupName=;   
GroupDomain=CN=,OU=,DC=; GroupRole=No Access;   
 GroupLoginRule=System.Object[]; GroupLoginInterface=System.Object[]},   
@{MemberId=2; GroupName=; GroupDomain=CN=,OU=,DC=;   
 GroupRole=No Access; GroupLoginRule=System.Object[];   
GroupLoginInterface=System.Object[]}, @{MemberId=3; GroupName=;   
 GroupDomain=CN=,OU=,DC=; GroupRole=No Access; GroupLoginRule=System.Object[];   
GroupLoginInterface=System.Object[]}...}

Output Description

For details, see [Parameters](#d0e21536).

## Modifying LDAP Information

Function

Modify LDAP information.

Format

**Set-iBMCLDAP** **-Session** *<$session>* **-LDAPID** *<LDAPID>* **-LDAPAddress** *<LDAPAddress>* **-LDAPPort** *<LDAPPort>* **-UserDomain** *<UserDomain>* **-BindDN** *<BindDN>* **-BindPassword** *<BindPassword>* **-CertificateVerificationEnabled** *<CertificateVerificationEnabled>* **-CertificateVerificationLevel** *<CertificateVerificationLevel>* **-GroupID** *<GroupID>* **-GroupName** *<GroupName>* **-GroupDomain** *<GroupDomain>* **-GroupRole** *<GroupRole>* **-GroupLoginRule** *<GroupLoginRule>* **-GroupLoginInterface** *<GroupLoginInterface>*

Parameters

| Parameter | Parameters | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<LDAPID>* | Specifies the ID of the LDAP resource. This parameter is mandatory. | An integer ranging from 1 to 6. |
| *<LDAPAddress>* | Specifies the address of the domain controller. This parameter is optional. | The value can be an IPv4 or IPv6 address, or a domain name. |
| *<LDAPPort>* | Specifies the port number of the domain controller. This parameter is optional. | An integer ranging from 1 to 65535. |
| *<UserDomain>* | Specifies the user domain of the domain controller. This parameter is optional. | Character string, for example:  CN=test,,DC=xfusion,DC=com |
| *<BindDN>* | Specifies the Distinguished name (DN) of an LDAP proxy user. This parameter is optional. | A string of 0 to 255 characters. |
| *<BindPassword>* | Specifies the authentication password of the LDAP proxy user. This parameter is optional. | A string of 0 to 20 characters. |
| *<CertificateVerificationEnabled>* | Specifies whether the certificate verification is enabled. This parameter is optional. | * **$true** or **1**: enabled * **$false** or **0**: disabled |
| *<CertificateVerificationLevel>* | Specifies the certificate verification level. This parameter is valid only when the certificate verification is enabled. | * Demand * Allow |
| *<GroupID>* | Specifies the ID of an LDAP user group. This parameter is optional. | An integer ranging from 0 to 4. |
| *<GroupName>* | Specifies the name of an LDAP user group. This parameter is optional. | Character string |
| *<GroupDomain>* | Specifies the domain of an LDAP user group. This parameter is optional. | Character string, for example:  CN=qwert,OU=admin,DC=xfusion,DC=com |
| *<GroupRole>* | Specifies the role of an LDAP user group. This parameter is optional. | * Administrator * Operator * Commonuser * Noaccess * CustomRole1 * CustomRole2 * CustomRole3 * CustomRole4 |
| *<GroupLoginRule>* | Specifies the login rules that apply to an LDAP user group. This parameter is optional. | One or more of **Rule1**, **Rule2**, and **Rule3**. |
| *<GroupLoginInterface>* | Specifies the interfaces through which the LDAP group members can log in to iBMC. | One or more of **Web**, **SSH**, and **Redfish**. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Modify LDAP information.

PS C:\> **$BindPwd = ConvertTo-SecureString -String bind-password -AsPlainText -Force**   
PS C:\> **$GroupLoginRule = ,@("Rule1", "Rule2", "Rule3")**   
PS C:\> **$GroupLoginInterface = ,@("Web","SSH","Redfish")**   
PS C:\> **$result = Set-iBMCLDAP -Session $Session -LDAPID 1 -LDAPAddress "ldap.xfusion.com" -LDAPPort 635 -UserDomain 'CN=test,,DC=xfusion,DC=com' -BindDN test -BindPassword $BindPwd -CertificateVerificationEnabled $true -CertificateVerificationLevel Allow -GroupID 0 -GroupName qwert -GroupDomain 'CN=qwert,OU=admin,DC=xfusion,DC=com' -GroupRole Administrator -GroupLoginRule $GroupLoginRule -GroupLoginInterface $GroupLoginInterface**

## Importing an LDAP certificate to the iBMC

Function

This command is used to import an LDAP certificate to the iBMC.

Format

**Import-iBMCLDAPCert** **-Session** *<$session>* **-LDAPID** *<LDAPID>* **-LDAPCertPath** *<LDAPCertPath>* **-SecureEnabled**

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | (Mandatory) Session | - |
| *<LDAPID>* | (Mandatory) ID of the LDAP resource | An integer ranging from 1 to 6. |
| *<LDAPCertPath>* | (Mandatory) Path for importing the certificate | * If a local directory is used, enter the path in the *Directory\File name* or *\\Client IP address\Directory\File name* format. * If a temporary directory of the iBMC is used, enter the path in the **/tmp**/*File name* format. * Using a remote directory:   + - When **-SecureEnabled** is configured, the dialog box for entering the user name and password of the remote protocol is displayed. The format of the file path is *File transfer protocol://Remote server IP address/Directory/File name*.     - When **-SecureEnabled** is not configured, the format of the file path is *File transfer protocol://User name:Password@Remote server IP address/Directory/File name*.     - File transfer protocols include HTTPS, SCP, SFTP, CIFS, and NFS. |
| -SecureEnabled | (Optional) Whether to enable the interactive function. The function is enabled when this parameter is added to the command. You are advised to enable this function when using a remote directory to ensure operation security. | - |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Import an LDAP certificate to the iBMC.

* Import an LDAP certificate from a local directory.

PS C:\> **$Result = Import-iBMCLDAPCert -Session $Session -LDAPID 1 -LDAPCertPath 'c:\ca.cer'**   
PS C:\> **$Result**   
   
Host : 10.1.1.2   
MessageId : iBMC.1.0.LDAPCertImportSuccess   
RelatedProperties :   
Message : The LDAP certificate is imported successfully.   
MessageArgs :   
Severity : OK   
Resolution : None

* Import an LDAP certificate from a remote server to the iBMC.

PS C:\> **$Result = Import-iBMCLDAPCert -Session $Session -LDAPID 1 -LDAPCertPath 'nfs://10.1.1.100/data/ldap.cer"'**   
PS C:\> **$Result**   
   
Host : 10.1.1.2   
Id : 1   
Name : ldap root cert import   
ActivityName : [10.1.1.2] ldap root cert import   
TaskState : Completed   
StartTime : 2019-11-28T16:44:46+08:00   
EndTime : 2019-11-28T16:44:47+08:00   
TaskStatus : OK   
TaskPercent : 100%

Output Description

Output information (import from a local PC)

| Parameter | Description |
| --- | --- |
| Host | Host name |
| MessageId | Message ID |
| RelatedProperties | Properties related to the message |
| Message | Message content |
| MessageArgs | Message arguments |
| Severity | Severity level |
| Resolution | Solution |

Output information (import from a remote server)

| Parameter | Description |
| --- | --- |
| Host | Host name |
| Id | Task ID |
| Name | Task name |
| ActivityName | Name of the task activity |
| TaskState | Task progress |
| StartTime | Start time of a task |
| EndTime | End time of a task |
| TaskStatus | Task execution status |
| TaskPercent | Task progress |

## Setting the LDAP Status

Function

This command is used to enable or disable the LDAP function.

Format

**Set-iBMCLDAPServiceEnabled** **-Session** *<$session>* **-LdapServiceEnabled** *<LdapServiceEnabled>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | (Mandatory) Session | - |
| *<LdapServiceEnabled>* | (Mandatory) Whether to enable the LDAP function. | * **$true**/**1**: enable the LDAP function. * **$false**/**0**: disable the LDAP function. |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Enable the LDAP function.

PS C:\> **$User = Set-iBMCLDAPServiceEnabled -Session $session -LdapServiceEnabled $true**   
PS C:\> **$User**   
   
Host : 10.1.1.2   
Id : 12   
Name : User Account   
LdapServiceEnabled : true

Output Description

Output information

| Parameter | Description |
| --- | --- |
| Host | Host name |
| ID | Task ID |
| Name | Task name |
| LdapServiceEnabled | LDAP service status |

## Querying Kerberos Service Resources

Function

Query Kerberos service resource information.

Format

**Get-iBMCKerberos -Session** *<$session>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |

Usage Guidelines

iBMC V601 or later is required.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query Kerberos service resource information.

PS C:\> **$System = Get-iBMCKerberos -Session $session**   
PS C:\> **$System**   
   
Host : 192.168.1.1   
Id : KerberosService   
Name : Kerberos Service   
KerberosEnabled : True

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the name of a host. |
| Id | Specifies the ID of a Kerberos resource. |
| Name | Specifies the name of a Kerberos resource. |
| KerberosEnabled | Specifies the Kerberos function status. The values are:   * **True**: The Kerberos function is enabled. * **False**: The Kerberos function is disabled. |

## Changing the Kerberos Enabling Status

Function

Modify the enabling status of the Kerberos function.

Format

**Set-iBMCKerberos -Session** *<$session>* **-KerberosEnabled** *<KerberosEnabled>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<KerberosEnabled>* | Specifies whether to enable the Kerberos function. This parameter is mandatory. | * **$true** or **1**: The Kerberos function is enabled. * **$false** or **0**: The Kerberos function is disabled. |

Usage Guidelines

iBMC V601 or later is required.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Change the enabling status of the Kerberos function.

PS C:\> **$System = Set-iBMCKerberos -Session $session -KerberosEnabled $True**   
PS C:\> **$System**   
   
Host : 192.168.1.1   
Id : KerberosService   
Name : Kerberos Service   
KerberosEnabled : True

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the name of a host. |
| Id | Specifies the ID of a Kerberos resource. |
| Name | Specifies the name of a Kerberos resource. |
| KerberosEnabled | Specifies the Kerberos function status. The values are:   * **True**: The Kerberos function is enabled. * **False**: The Kerberos function is disabled. |

## Querying the Kerberos Domain Controller Information

Function

Query the information about a Kerberos domain controller.

Format

**Get-iBMCKerberosController -Session** *<$session>* **-Id** *<Id>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | (Mandatory) Session | - |
| *<Id>* | (Optional) Domain controller ID, obtained from the queried Kerberos domain controller resources. | The value can only be **1**. |

Usage Guidelines

iBMC V601 or later is required.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the information about a Kerberos domain controller.

PS C:\> **$System = Get-iBMCKerberosController -Session $session** **-Id 1**   
PS C:\> **$System**   
   
Host : 192.168.1.1   
Name : Kerberos Controller   
Id : 1   
KerberosServerAddress : hi   
KerberosPort : 88   
Realm : realm   
KerberosGroups : {@{MemberId=0; GroupName=testluzhou; GroupDomain=hello; GroupSID=; GroupRole=No Access; GroupLo   
 ginRule=System.Object[]; GroupLoginInterface=System.Object[]}, @{MemberId=1; GroupName=test2; G   
 roupDomain=chedck world; GroupSID=test2; GroupRole=Common User; GroupLoginRule=System.Object[];   
 GroupLoginInterface=System.Object[]}, @{MemberId=2; GroupName=testluzhou; GroupDomain=CN=12345   
 ; GroupSID=5678; GroupRole=Administrator; GroupLoginRule=System.Object[]; GroupLoginInterface=S   
 ystem.Object[]}, @{MemberId=3; GroupName=; GroupDomain=; GroupSID=; GroupRole=No Access; GroupL   
 oginRule=System.Object[]; GroupLoginInterface=System.Object[]}...}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Host name |
| Id | ID of a Kerberos resource |
| Name | Name of a Kerberos resource |
| KerberosServerAddress | Address of a domain controller |
| KerberosPort | Port number of a domain controller |
| Realm | Domain of a domain controller. |
| KerberosGroups | Kerberos user group. A maximum of five Kerberos user groups are supported. |
| MemberId | ID of a Kerberos user group |
| GroupName | Name of a Kerberos user group |
| GroupDomain | Group domain of a Kerberos user group |
| GroupSID | SID of a Kerberos user group |
| GroupRole | Role of a Kerberos user group |
| GroupLoginRule | Login rules applied to a Kerberos user group. Each item is an object. The attribute in the object is as follows:  **@odata.id**: path to a specific login rule. |
| GroupLoginInterface | Login interface of a Kerberos user group. Each item is a character string. Currently, only WebUI login is supported. |

## Modifying the Kerberos Domain Controller Information

Function

This command is used to modify the information of a specific Kerberos domain controller.

Format

**Set-iBMCKerberosController –Session** *<$session>* **-Id** *<Id>* **-SeverAddress** *<SeverAddress>* **-Port** *<Port>* **-Realm** *<Realm>* **-GroupName** *<GroupName>* **-GroupDomain** *<GroupDomain>* **-GroupSID** *<GroupSID>* **-GroupRole** *<GroupRole>* **-RuleId** *<RuleId>* **-InterId** *<InterId>* **-GroupID** *<GroupID>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | (Mandatory) Session | - |
| *<Id>* | (Optional) Domain controller ID, obtained from the queried Kerberos domain controller resources. | The value can only be **1**. |
| *<SeverAddress>* | (Optional) Address of a domain controller, which can be an IPv4 or IPv6 address, or a domain name. | - |
| *<Port>* | (Optional) Port number of a domain controller. The default value is **88**. | The value is an integer ranging from **1** to **65535**. |
| *<Realm>* | (Optional) Domain name of a domain controller, which must be capitalized, for example, **TEST.XFUSION.COM**. | - |
| *<GroupName>* | (Optional) Name of a Kerberos user group. A maximum of 255 characters is supported. | - |
| *<GroupDomain>* | (Optional) Group domain of a Kerberos user group. A maximum of 255 characters is supported. | - |
| *<GroupSID>* | (Optional) SID of a Kerberos user group. A maximum of 255 characters is supported.  NOTE  Setting this parameter to null or a null string ('' or "") indicates that the configuration information about the group is deleted. | - |
| *<GroupRole>* | (Optional) Role of a Kerberos user group | * Common User * Operator * Administrator * Custom Role 1 * Custom Role 2 * Custom Role 3 * Custom Role 4 * No Access |
| *<RuleId>* | (Optional) Login rules applied to a Kerberos user group. The value is an array consisting of character strings, for example, **["Rule1","Rule2","Rule3"]**. | - |
| *<InterId>* | (Optional) Login interface of a Kerberos user group. Currently, only WebUI login is supported. | Currently, this parameter can only be set to **["Web"]**. |
| *<GroupID>* | (Optional) ID of a Kerberos user group. | The value can be **0**, **1**, **2**, **3**, or **4**. |

Usage Guidelines

iBMC V601 or later is required.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Modify the information of a specific Kerberos domain controller.

PS C:\> **$System = Set-iBMCKerberosController -Session $session -Id 1 -Port 89**   
PS C:\> **$System**   
   
Host : 192.168.1.1   
Id : 1   
Name : Kerberos Controller   
KerberosServerAddress : check   
KerberosPort : 89   
Realm : here   
KerberosGroups : {@{MemberId=0; GroupName=; GroupDomain=; GroupSID=; GroupRole=No Access; GroupLoginRule=System.   
 Object[]; GroupLoginInterface=System.Object[]}, @{MemberId=1; GroupName=; GroupDomain=; GroupSI   
 D=; GroupRole=No Access; GroupLoginRule=System.Object[]; GroupLoginInterface=System.Object[]},   
 @{MemberId=2; GroupName=; GroupDomain=; GroupSID=; GroupRole=No Access; GroupLoginRule=System.O   
 bject[]; GroupLoginInterface=System.Object[]}, @{MemberId=3; GroupName=; GroupDomain=; GroupSID   
 =; GroupRole=No Access; GroupLoginRule=System.Object[]; GroupLoginInterface=System.Object[]}...   
 }

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Host name |
| Id | ID of a Kerberos resource |
| Name | Name of a Kerberos resource |
| KerberosServerAddress | Address of a domain controller |
| KerberosPort | Port number of a domain controller |
| Realm | Domain of a domain controller. |
| KerberosGroups | Kerberos user group. A maximum of five Kerberos user groups are supported. |
| MemberId | ID of a Kerberos user group |
| GroupName | Name of a Kerberos user group |
| GroupDomain | Group domain of a Kerberos user group |
| GroupSID | SID of a Kerberos user group |
| GroupRole | Role of a Kerberos user group |
| GroupLoginRule | Login rules applied to a Kerberos user group. Each item is an object. The attribute in the object is as follows:  **@odata.id**: path to a specific login rule. |
| GroupLoginInterface | Login interface of a Kerberos user group. Each item is a character string. Currently, only WebUI login is supported. |

## Importing a Kerberos Domain Controller Key Table

Function

This command is used to import the key table of the Kerberos domain controller.

Format

**Import-iBMCKerberosControllerKeyTable –Session** *<$session>* **-Id** *<Id>* **-KeyTable** *<KeyTable>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | (Mandatory) Session | - |
| *<Id>* | (Mandatory) ID of a domain controller, which is obtained from the queried Kerberos domain controller resources. The value is an integer array. | - |
| *<KeyTable>* | (Mandatory) Local path of a key table | * If a local directory is used, enter the path in the *Directory\File name* or *\\Client IP address\Directory\File name* format. * If a temporary directory of the iBMC is used, enter the path in the **/tmp**/*File name* format. |

Usage Guidelines

iBMC V601 or later is required.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Import the key table of the Kerberos domain controller.

PS C:\> **$result = Import-iBMCKerberosControllerKeyTable -Session $session -Id 1 -KeyTable "C:\Users\test\Desktop\iBMC1021.keytab"**   
 PS C:\> **$result**   
   
Host : 192.168.1.1   
MessageId : iBMC.1.0.KRBKeytabUploadSuccess   
RelatedProperties : {}   
Message : The Kerberos key table is uploaded successfully.   
MessageArgs : {}   
Severity : OK   
Resolution : None

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Host name |
| MessageId | Message ID |
| RelatedProperties | Properties related to the message |
| Message | Message content |
| MessageArgs | Message arguments |
| Severity | Severity, which can be **OK**, **Warning**, or **Critical**. |
| Resolution | Solution |

## Querying a Diagnostic Service Resource

Function

This command is used to query the diagnostic service resource.

Format

**Get-iBMCDiagnosticService -Session** *<$session>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | (Mandatory) Session | - |

Usage Guidelines

iBMC V550 or later is required.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the diagnostic service resource.

PS C:\> **$DiagnosticServices = Get-iBMCDiagnosticService -Session $session**   
 PS C:\> **$DiagnosticServices**   
   
 Host : 10.1.1.2   
 VideoRecordingEnabled : True   
 ScreenShotEnabled : True   
 BlackBoxEnabled : True   
 SerialPortDataEnabled : True   
 VideoPlaybackConnNum : 0   
 VideoRecordInfo : {$null, @{VideoSizeByte=324404; CreateTime=2021-02-23 13:22:21}, @{VideoSizeByte=606024; Cr   
 eateTime=2021-02-26 07:21:07}}   
 ScreenShotCreateTime : {2021-02-26 07:20:44, 2021-02-26 07:08:48, 2021-02-26 07:09:11, $null}

Output Description

Output description

| Output | Description |
| --- | --- |
| Host | Host name |
| VideoRecordingEnabled | Status of the recording function |
| ScreenShotEnabled | Status of the last-screen screenshot function |
| BlackBoxEnabled | Status of the black box function |
| SerialPortDataEnabled | Status of the function of recording system serial port data |
| VideoPlaybackConnNum | Number of recording playback channels |
| VideoRecordInfo | Recording information |
| ScreenShotCreateTime | Screenshot creation time |

## Modifying Diagnostic Service Resource Information

Function

This command is used to modify diagnostic service resource information.

Format

**Set-iBMCDiagnosticService** **-Session** *<$session>* **-VideoRecordingEnabled** *<VideoRecordingEnabled>* **-ScreenShotEnabled** *<ScreenShotEnabled>* **-BlackBoxEnabled** *<BlackBoxEnabled>* **-SerialPortDataEnabled** *<SerialPortDataEnabled>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | (Mandatory) Session | - |
| *<VideoRecordingEnabled>* | (Optional) Whether to enable the recording function. | * **$true**/**1**: enable the recording function. * **$false**/**0**: disable the recording function. |
| *<ScreenShotEnabled>* | (Optional) Whether to enable the last-screen screenshot function. | * **$true**/**1**: enable this function. * **$false**/**0**: disable this function. |
| *<BlackBoxEnabled>* | (Optional) Whether to enable the black box function. | * **$true**/**1**: enable this function. * **$false**/**0**: disable this function. |
| *<SerialPortDataEnabled>* | (Optional) Whether to enable the function of recording system serial port data. | * **$true**/**1**: enable this function. * **$false**/**0**: disable this function. |

Usage Guidelines

iBMC V550 or later is required.

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Configure diagnosis service resources.

PS C:\> **Set-iBMCDiagnosticService -Session $session -VideoRecordingEnabled $true -ScreenShotEnabled $true -BlackBoxEnabled $true -SerialPortDataEnabled $true**

## General Interface

Function

Provide a general interface.

Format

**Invoke-iBMCGeneralCall -Session** *<$session>* **-url** *<url>* **-method** *<method>* **-ContentType** <contentType> **-Payload** *<payload>* **-file** *<filePath>*

Parameters

| Parameter | Description | Value |
| --- | --- | --- |
| *<$session>* | Specifies the session. This parameter is mandatory. | - |
| *<url>* | Specifies the URL of a request. This parameter is mandatory. | - |
| *<method>* | Specifies the type of a request. This parameter is mandatory. | * patch * post * get * delete |
| <contentType> | Specifies the format of a request message. This parameter is mandatory. | * Form * Json |
| *<payload>* | Specifies the message body of a request, which is a hash table array.   * This parameter is mandatory when the request type is **post** or **patch**. You can set either *<payload>* or *<filePath>*. * This parameter is optional when the request type is **get** or **delete**. | - |
| *<filePath>* | Specifies the local path for a response. JSON files or other files are supported.   * This parameter is mandatory when the request type is **post** or **patch**. You can set either *<payload>* or *<filePath>*. * This parameter is optional when the request type is **get** or **delete**. | - |

Usage Guidelines

None

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Provide a general interface.

PS C:\> **$System = Invoke-iBMCGeneralCall -Session $session -Url "/AccountService/KerberosService" -Method 'Get' -ContentType Json**   
PS C:\> **$System**   
   
Host : 192.168.1.1   
@odata.context : /redfish/v1/$metadata#AccountService/KerberosService   
@odata.id : /redfish/v1/AccountService/KerberosService   
@odata.type : #HwKerberosService.v1\_0\_0.HwKerberosService   
Id : KerberosService   
Name : Kerberos Service   
KerberosEnabled : True   
KerberosControllers : @{@odata.id=/redfish/v1/AccountService/KerberosService/KerberosControllers}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Specifies the name of a host. |
| @odata.context | Specifies the OData information of a task service resource model. |
| @odata.id | Specifies the access path of the current task resource. |
| @odata.type | Specifies the type of a task service resource. |
| Id | Specifies the ID of a Kerberos resource. |
| Name | Specifies the name of a Kerberos resource. |
| KerberosEnabled | Specifies the enabling status of the Kerberos function. |
| KerberosControllers | Specifies the address of a Kerberos domain controller. The value is as follows:   * **@odata.id**: the domain controller path |

## Importing the Root Certificate of a Remote HTTPS Server

Function

The root certificate of the remote HTTPS server can be imported to the local path, the **tmp** directory of the iBMC, or the remote path.

Format

**Import-iBMCRemoteHttpsServerRootCA** **-Session** <*$session* > **-CertPath** *<CertPath>* **-CertID** *<CertID>***-Usage** *<Usage>* **-SecureEnabled**

Parameters

| Parameter | Description | Specifications |
| --- | --- | --- |
| *<$session>* | (Mandatory) Session. | - |
| *<CertPath>* | The path of the root certificate of the remote HTTPS server.  The local path, the **tmp** directory of the iBMC, and the remote path are supported. The file name extension must be **.crt**,**.cer** or **.pem**. | * If a local directory is used, enter the path in the *Directory\File name* or *\\Client IP address\Directory\File name* format. * If a temporary directory of the iBMC is used, enter the path in the **/tmp**/*File name* format. * Using a remote directory:   + - When **-SecureEnabled** is configured, the dialog box for entering the user name and password of the remote protocol is displayed. The format of the file path is *File transfer protocol://Remote server IP address/Directory/File name*.     - When **-SecureEnabled** is not configured, the format of the file path is *File transfer protocol://User name:Password@Remote server IP address/Directory/File name*.     - File transfer protocols include HTTPS, SCP, SFTP, CIFS, and NFS. |
| *<CertID>* | ID of the root certificate authenticated by the remote HTTPS server.  The value is an integer that ranges from 5 to 8. Certificates 5 to 8 are used for remote file transfer over HTTPS.  NOTE  Either **CertID** or **usage** must be set. | 5~8 |
| *<Usage>* | Certificate usage.  NOTE  Either **CertID** or **usage** must be set. | FileTransfer |
| -SecureEnabled | This parameter is optional and specifies whether to enable the interactive input of sensitive information on the file server. If this parameter is added to the command, the interactive input function is enabled. You are advised to enable this function when using a remote directory to ensure operation security. | - |

Usage Guidelines

Only the following iBMC versions support this function:

* iBMC 3.01.12.20 or later

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Import the root certificate of a remote HTTPS server.

PS C:\> **$Tasks = Import-iBMCRemoteHttpsServerRootCA $session 'sftp://192.168.1.2/data/https.crt' -CertID 5 -SecureEnabled**   
   
Host : 192.168.1.1   
Id : 2   
Name : remote https server ca import   
ActivityName : [192.168.1.1] remote https server ca import   
TaskState : Completed   
StartTime : 2020-11-14T17:54:54+08:00   
EndTime : 2020-11-14T17:56:06+08:00   
TaskStatus : OK   
TaskPercent : 100%

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Host name |
| Id | Task ID |
| Name | Task name |
| ActivityName | Task name |
| TaskState | Task progress |
| StartTime | Task start time |
| EndTime | Task end time |
| TaskStatus | Task status |
| TaskPercent | Task completion rate |

## Deleting the Root Certificate of a Remote HTTPS Server

Function

Delete the root certificate of a remote HTTPS server.

Format

**Remove-iBMCRemoteHttpsServerRootCA** **-Session** *<$session>* **-CertID** *<CertID>*

Parameters

| Parameter | Description | Specifications |
| --- | --- | --- |
| *<$session>* | (Mandatory) Session. | - |
| *<CertID>* | ID of the root certificate authenticated by the remote HTTPS server.  The value is an integer that ranges from 5 to 8. Certificates 5 to 8 are used for remote file transfer over HTTPS. | 5~8 |

Usage Guidelines

Only the following iBMC versions support this function:

* iBMC 3.01.12.20 or later

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Delete the root certificate of a remote HTTPS server.

PS C:\> **Remove-iBMCRemoteHttpsServerRootCA -Session $session -CertID 5**

## Importing a CRL of a Remote HTTPS Server

Function

The CRL of the root certificate of the remote HTTPS server can be imported to the local path, the **tmp** directory of the iBMC, or the remote path.

Format

**Import-iBMCRemoteHttpsServerCrl** **-Session** <*$session* > **-CertPath** *<CertPath>* **-CertID** *<CertID>***-Usage** *<Usage>* **-SecureEnabled**

Parameters

| Parameter | Description | Specifications |
| --- | --- | --- |
| *<$session>* | (Mandatory) Session. | - |
| *<CertPath>* | The path of the CRL of the root certificate of the remote HTTPS server.  The local path, the **tmp** directory of the iBMC, and the remote path are supported. The file name extension must be **.crt**,**.cer** or **.pem**. | * If a local directory is used, enter the path in the *Directory\File name* or *\\Client IP address\Directory\File name* format. * If a temporary directory of the iBMC is used, enter the path in the **/tmp**/*File name* format. * Using a remote directory:   + - When **-SecureEnabled** is configured, the dialog box for entering the user name and password of the remote protocol is displayed. The format of the file path is *File transfer protocol://Remote server IP address/Directory/File name*.     - When **-SecureEnabled** is not configured, the format of the file path is *File transfer protocol://User name:Password@Remote server IP address/Directory/File name*.     - File transfer protocols include HTTPS, SCP, SFTP, CIFS, and NFS. |
| *<CertID>* | ID of the root certificate object that issues the CRL.  The value must be the **CertId** value of an array member in the **RootCertificate** object returned when the user service information is queried. The value is an integer that ranges from 5 to 8. Certificates 5 to 8 are used for remote file transfer over HTTPS.  NOTE  Either **CertID** or **usage** must be set. | 5~8 |
| *<Usage>* | Certificate usage.  NOTE  Either **CertID** or **usage** must be set. | FileTransfer |
| -SecureEnabled | This parameter is optional and specifies whether to enable the interactive input of sensitive information on the file server. If this parameter is added to the command, the interactive input function is enabled. You are advised to enable this function when using a remote directory to ensure operation security. | - |

Usage Guidelines

Only the following iBMC versions support this function:

* iBMC 3.01.12.20 or later

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Import the CRL of the remote HTTPS server.

PS C:\> **$Tasks = Import-iBMCRemoteHttpsServerCrl $session 'sftp://192.168.1.2/data/https.crl' -CertID 5 -SecureEnabled**   
   
Host : 192.168.1.1   
Id : 2   
Name : remote https server crl import   
ActivityName : [192.168.1.1] remote https server crl import   
TaskState : Completed   
StartTime : 2020-11-14T17:54:54+08:00   
EndTime : 2020-11-14T17:56:06+08:00   
TaskStatus : OK   
TaskPercent : 100%

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Host name |
| Id | Task ID |
| Name | Task name |
| ActivityName | Task name |
| TaskState | Task progress |
| StartTime | Task start time |
| EndTime | Task end time |
| TaskStatus | Task status |
| TaskPercent | Task completion rate |

## Querying the Certificate Verification Switch Status of the Remote HTTPS Server

Function

Query the certificate verification switch status of the remote HTTPS server.

Format

**Get-iBMCSecurityService** **-Session** *<$session>*

Parameters

| Parameter | Description | Specifications |
| --- | --- | --- |
| *<$session>* | (Mandatory) Session. | - |

Usage Guidelines

Only the following iBMC versions support this function:

* iBMC 3.01.12.20 or later

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Query the certificate verification switch status of the HTTPS server.

PS C:\> **$Result = Get-iBMCSecurityService -Session $Session**   
PS C:\> **$Result | fl**   
   
Host : 192.168.1.1   
Id : SecurityService   
Name : Security Service   
SessionTokenLength : 16   
SecurityControlVersion : 1   
MasterKeyUpdateInterval : 0   
HttpsTransferCertVerification : False   
SOLAutoOSLockEnabled : False   
SOLAutoOSLockKey : 0   
RemoteHttpsServerCertChainInfo : {}

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Host name |
| Id | ID of the security service collection |
| Name | Name of the security service collection |
| SessionTokenLength | Token length of a Redfish message |
| SecurityControlVersion | Versions that support the security feature |
| MasterKeyUpdateInterval | Master key update interval |
| HttpsTransferCertVerification | The certificate verification switch status of the remote HTTPS server   * True: enabled * False: disabled |
| SOLAutoOSLockEnabled | OS automatic screen locking status of SOL |
| SOLAutoOSLockKey | OS automatic screen lock button of SOL |
| RemoteHttpsServerCertChainInfo | Information about the certificate chain and CRL of the remote HTTPS server |

## Enabling/Disabling the Certificate Verification of the Remote HTTPS Server

Function

Enable or disable the certificate verification function of the HTTPS file server.

Format

**Set-iBMCSecurityService** **-Session** *<$session>* **-HttpsTransferCertVerification** *<HttpsTransferCertVerification>*

Parameters

| Parameter | Description | Specifications |
| --- | --- | --- |
| *<$session>* | (Mandatory) Session. | - |
| *<HttpsTransferCertVerification>* | Configure the certificate verification status of the remote HTTPS server. | * $true or 1: Enabled. * $false or 0: Disabled. |

Usage Guidelines

Only the following iBMC versions support this function:

* iBMC 3.01.12.20 or later

Example

1. Log in to the iBMC. For details, see 3.3 Connecting to the iBMC.
2. Configure the certificate verification status of the remote HTTPS server.

PS C:\> **$Result = Set-iBMCSecurityService -Session $Session -HttpsTransferCertVerification $true**   
PS C:\> **$Result**   
   
Host : 192.168.1.1   
Id : SecurityService   
Name : Security Service   
HttpsTransferCertVerification : true

Output Description

Output description

| Output Item | Description |
| --- | --- |
| Host | Host name |
| Id | ID of the security service collection |
| Name | Name of the security service collection |
| HttpsTransferCertVerification | The certificate verification switch status of the remote HTTPS server   * True or 1: enabled * False or 0: disabled |

# FAQs

[4.1 How to Use an Encrypted Password When Invoking Cmdlets](#_EN-US_TOPIC_0000001137285127)

[4.2 How to Use the CA Root Certificate to Authenticate a Server](#_EN-US_TOPIC_0000001137285169)

[4.3 How to Invoke a Cmdlets Interface](#_EN-US_TOPIC_0000001137285215)

[4.4 How to Record Logs as a Non-Administrator User](#_EN-US_TOPIC_0000001137285041)

## How to Use an Encrypted Password When Invoking Cmdlets

Generate a key and password file.

1. Generate a 32-bit key and save it to the **aes.key** file.

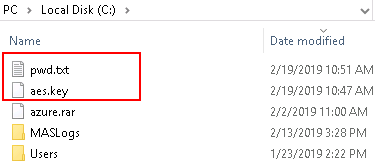
PS C:\Users\Administrator> **$KeyFile = "C:\aes.key"**   
PS C:\Users\Administrator> **$Key = New-Object Byte[] 32**   
PS C:\Users\Administrator> **[Security.Cryptography.RNGCryptoServiceProvider]::Create().GetBytes($Key)**

1. Use the key to generate a password file and save the file.

PS C:\Users\Administrator> **$Key | Out-File $KeyFile**   
PS C:\Users\Administrator> **Read-Host "Enter Password" -AsSecureString | ConvertFrom-SecureString -key $Key |Out-File "C:\pwd.txt"**   
Enter Password: \*\*\*\*\*\*\*\*\*\*   
PS C:\Users\Administrator>

1. You can obtain the **aes.key** and **pwd.txt** files from the specified directory in the environment. The encrypted key and password file can be used in any Cmdlets environment.

aes.key and pwd.txt files



Use the password file and key file to create credential information.

PS C:\Users\Administrator> **$userName = "root"**   
PS C:\Users\Administrator> **$passwdFile = "C:\pwd.txt"**   
PS C:\Users\Administrator> **$keyFile = "C:\aes.key"**   
PS C:\Users\Administrator> **$key = Get-Content $keyFile**   
PS C:\Users\Administrator> **$cred = New-Object -TypeName System.Management.Automation.PSCredential -ArgumentList $userName, (Get-Content $passwdFile | ConvertTo-SecureString -key $key)**

Use the credential information to set up a session.

PS C:\Users\Administrator> **Connect-iBMC -Address 10.10.10.1 -Credential $cred -TrustCert**   
   
Id : Blade4   
Name : Manager   
ManagerType : BMC   
FirmwareVersion : 3.00   
UUID : 14373629-DC99-A151-E811-F789DE92D0A2   
Model : iBMC   
Health : OK   
State : Enabled   
DateTime : 2019-02-19T03:43:33+00:00   
DateTimeLocalOffset : GMT   
Address : 10.10.10.1   
BaseUri : https://10.10.10.1   
Location : /redfish/v1/SessionService/Sessions/96c42b21c44822ea   
Alive : True   
AuthToken : 4e2c838c6f6514a6a1741e09a98b1330   
TrustCert : True

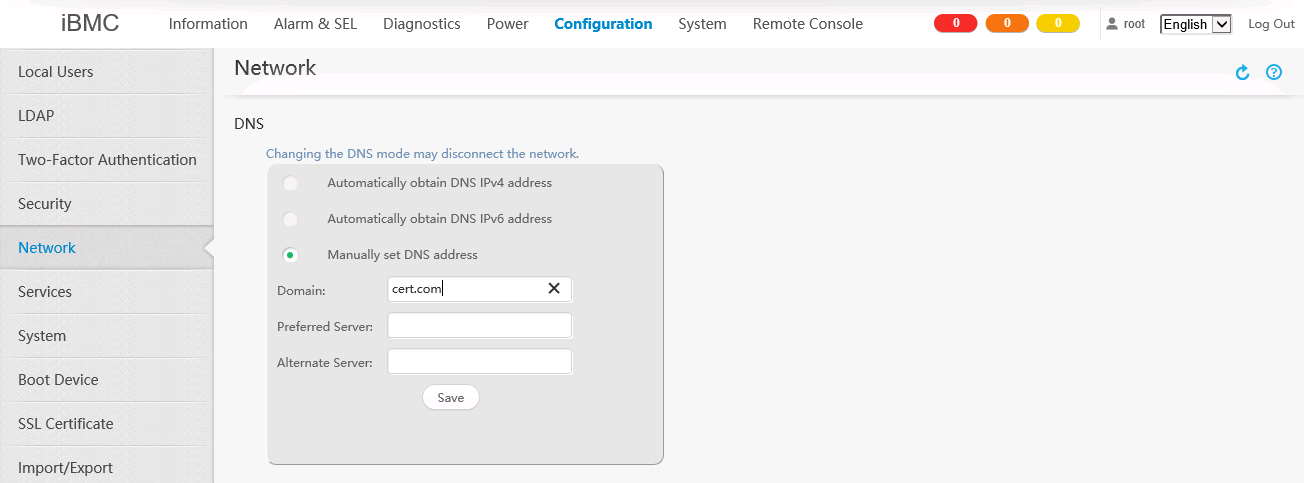
----End

## How to Use the CA Root Certificate to Authenticate a Server

Set the iBMC FQDN.

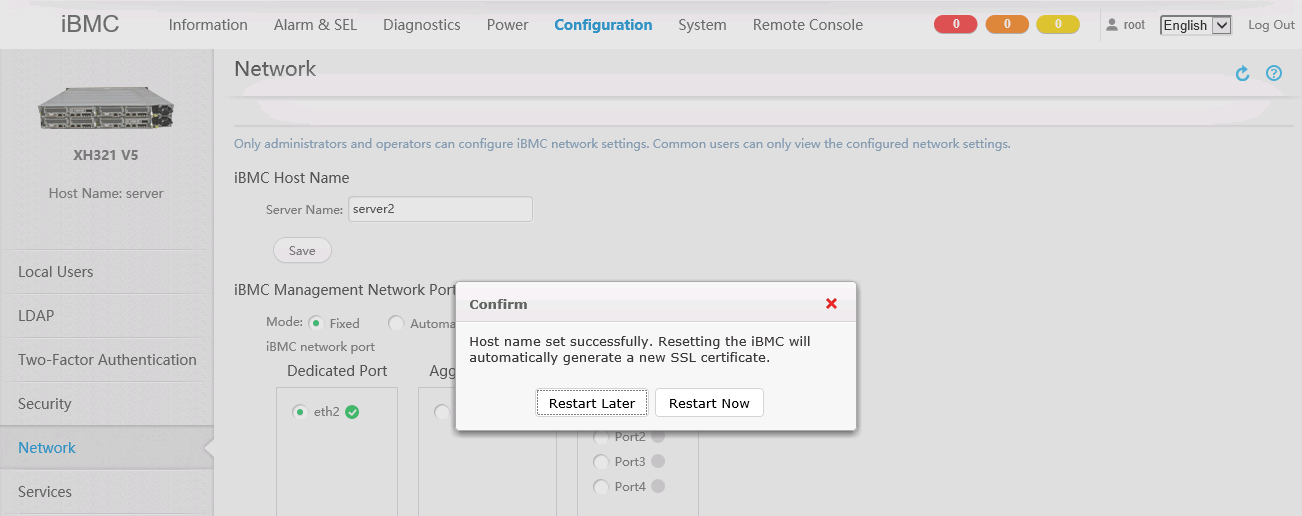
1. Log in to the iBMC WebUI, and choose **Configuration** > **Network**.
2. In the **DNS** area, set the domain name, for example, **cert.com**, as shown in Figure 4-2.

Setting the domain name



1. Click **Save**.
2. Set the iBMC host name, for example, **Server2**, as shown in Figure 4-3.

Setting the iBMC host name



1. Click **Save**. The **Confirm** dialog box is displayed.
2. Click **Restart Now** for the setting to take effect.

Configure the iBMC DNS.

1. Log in to the DNS server and configure the forward and reverse resolution for the iBMC.
2. Set the DNS server in [Step 2.1](#li2385450183919) as the DNS server in the Cmdlets environment.
3. In the Cmdlets environment, open a browser, enter **https://server2.cert.com** in the address box, and press **Enter** to log in to the iBMC.

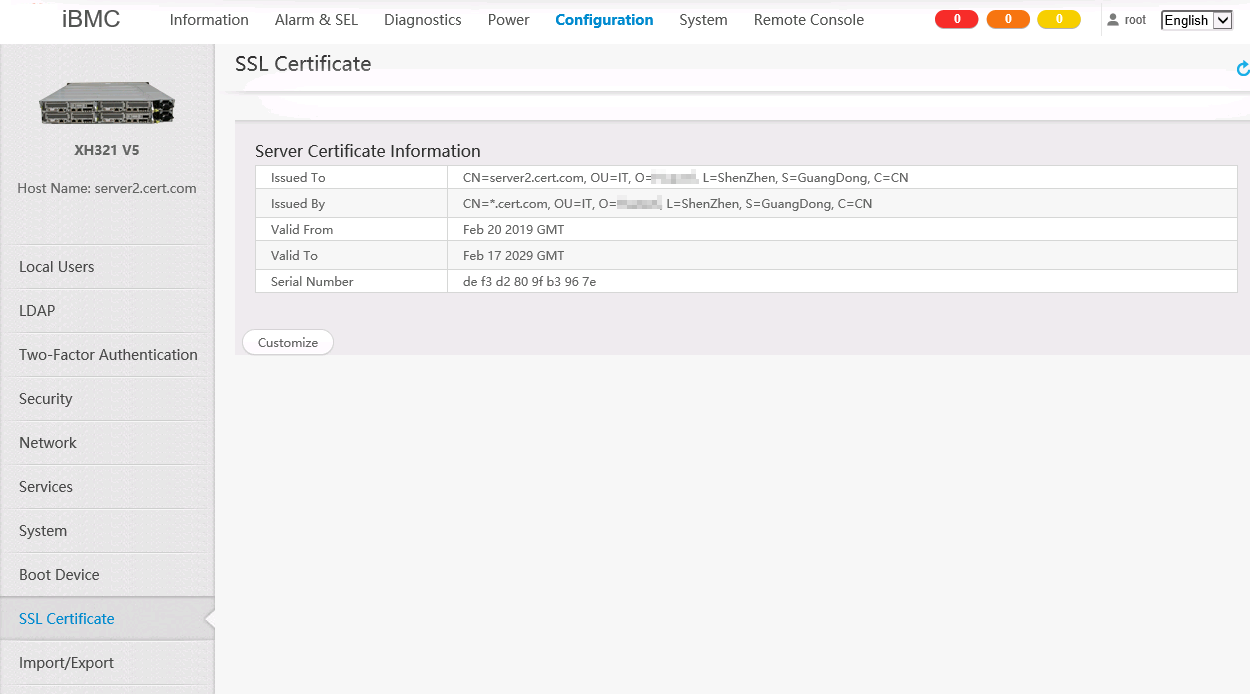
If the server iBMC WebUI is displayed, the binding is successful.

Generate an iBMC server certificate.

1. Generate a CSR file.
   1. Log in to the iBMC WebUI, and choose **Configuration** > **SSL Certificate**.

The **SSL Certificate** page is displayed, as shown in Figure 4-4.

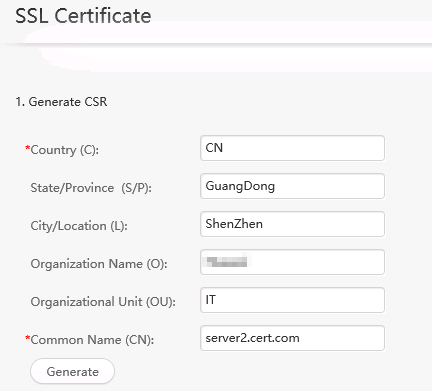
SSL Certificate page



* 1. Click **Customize**.

The customization page is displayed, as shown in Figure 4-5.

Customization page



* 1. Click **Generate**, and save the generated **server.csr** file to a local path.

1. Generate a server certificate using the CA root certificate and CSR file.

For example, use the OpenSSL tool to run the following command to generate a server certificate:

openssl x509 -req -days 3650 -sha256 -extensions server\_cert -extfile openssl.conf -CA ca.cer -CAkey ca.key -CAserial ca.srl -CAcreateserial -in server.csr -out server.cer

Import the CA root certificate to the Cmdlets running environment.



If the CA root certificate has been imported, you do not need to import it again.

1. Double-click the root certificate **ca.cer**.

The **Certificate** dialog box s displayed, as shown Figure 4-6.

Certificate

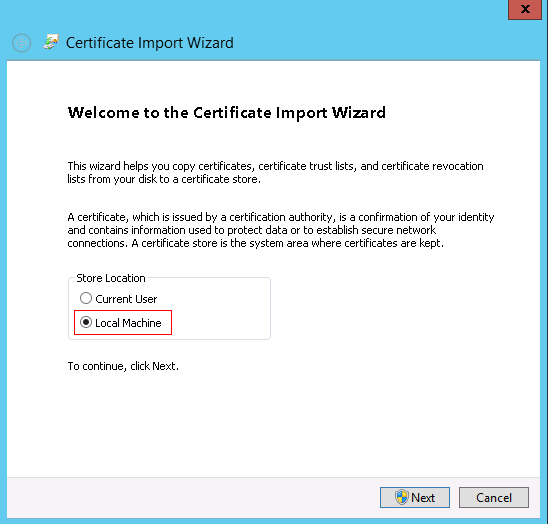


1. Click **Install Certificate**.

The **Certificate Import Wizard** dialog box is displayed.

1. Select **Local Machine**, as shown in Figure 4-7.

Setting Local Machine

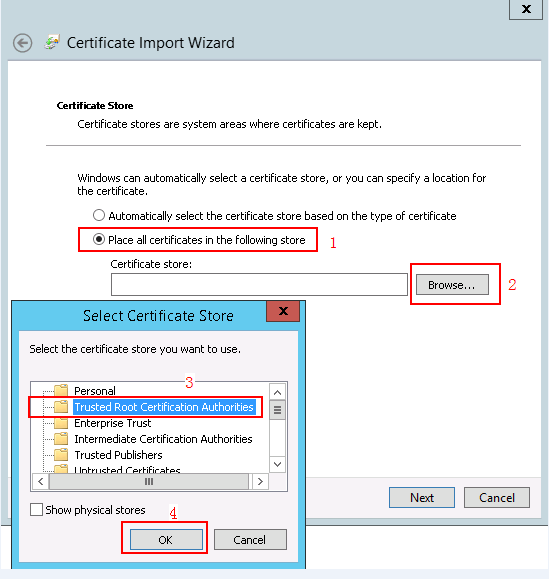


1. Click **Next**.

The **Certificate Store** dialog box is displayed.

1. Select **Place all certificates in the following store**, as shown by (1) in Figure 4-8.

Certificate Store



1. Click **Browse**, as shown by (2) in Figure 4-8.

The **Select Certificate Store** dialog box is displayed.

1. Select **Trusted Root Certification Authorities** and click **OK**, as shown by (3) and (4) in Figure 4-8.
2. Click **Next**.

The dialog box shown in Figure 4-9 is displayed.

Certificate imported



1. Click **Finish**.

The dialog box shown in Figure 4-10 is displayed.

Import successful



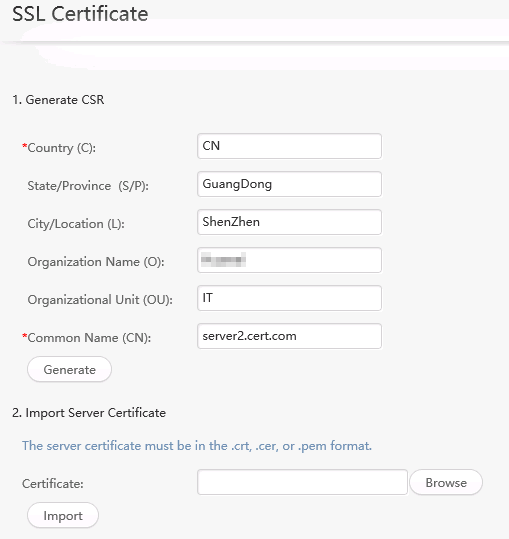
1. Click **OK**.
2. In the dialog box displayed as shown in Figure 4-6, click **OK**.

Import the iBMC server certificate.

1. Log in to the iBMC WebUI, and choose **Configuration** > **SSL Certificate** > **Customize**.

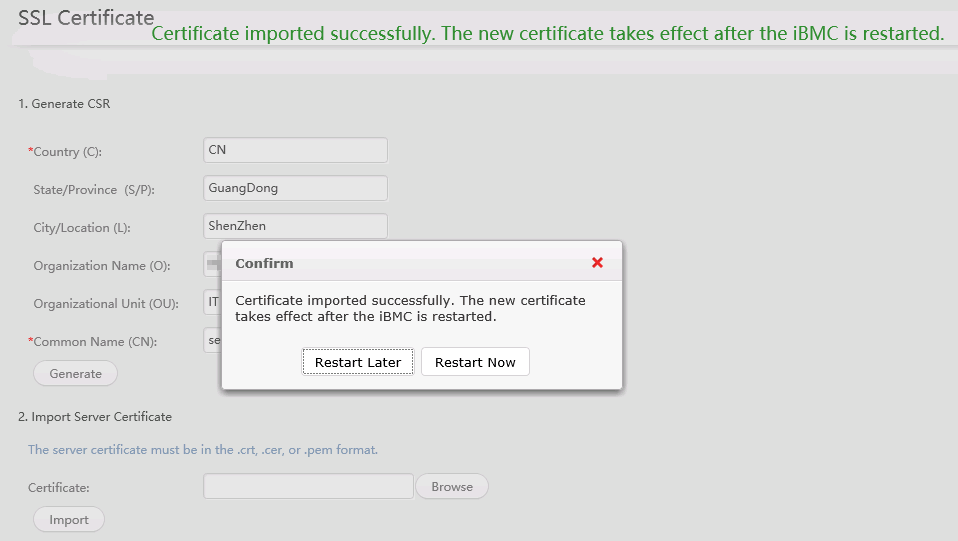
The customization page is displayed, as shown in Figure 4-11.

Customization page



1. Click **Browse** and select the **server.cer** file, as shown in Figure 4-12.

Importing the server certificate

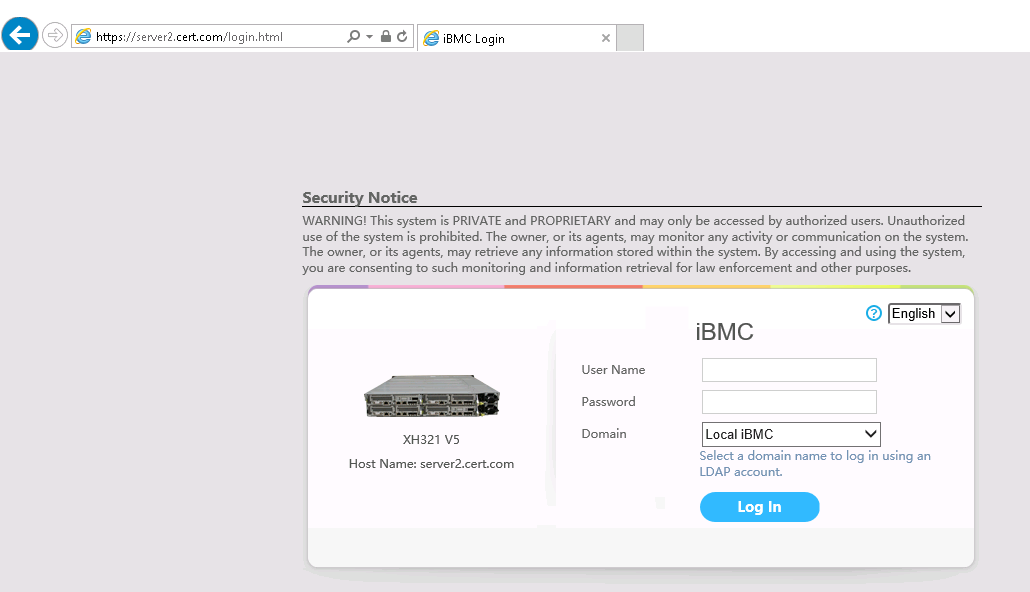


1. Click **Import**. The **Confirm** dialog box is displayed.
2. Click **Restart Now** for the setting to take effect.

Verify the result.

* In the Cmdlets environment, log in to the iBMC WebUI using the domain name. No alarm information about the certificate is displayed, as shown in Figure 4-13.

No alarm



* Use Cmdlets to create an iBMC connection without adding **-TrustCert**. The session is created successfully.

PS C:\Users\Administrator> $session = Connect-iBMC -Address server2.cert.com -Credential $cred   
PS C:\Users\Administrator> $session   
   
   
Id : Blade4   
Name : Manager   
ManagerType : BMC   
FirmwareVersion : 3.00   
UUID : 14373629-DC99-A151-E811-F789DE92D0A2   
Model : iBMC   
Health : OK   
State : Enabled   
DateTime : 2019-02-23T01:09:19+08:00   
DateTimeLocalOffset : Asia/Shanghai   
Address : server2.cert.com   
BaseUri : https://server2.cert.com   
Location : /redfish/v1/SessionService/Sessions/cdbc3a0503c9c670   
Alive : True   
AuthToken : 859a096a61462675213390d65306578a

----End

## How to Invoke a Cmdlets Interface

The following uses asset tag query and configuration as an example:

Import-Module -Name xFusion-iBMC-Cmdlets -Force   
   
try {   
 # connect to ibmc   
 $iBMCAddressList = @("10.1.1.1", "10.1.1.2")   
 $Sessions = Connect-iBMC -Address $iBMCAddressList -Username root -Password "xFusion@123" -TrustCert   
   
 # filter all invalid sessions   
 $InvaildSessions = @($($Sessions | Where-Object {$\_ -is [Exception]}))   
 # display session connect failed reason   
 if ($InvaildSessions.Count -gt 0) {   
 $InvaildSessions | ForEach-Object {   
 Write-Error -Exception $\_   
 }   
 return   
 }   
   
 # Codes start   
   
 ***# set asset tag***   
 ***$AssertTagList = @("server1", "server2")***   
 ***$SetResults = Set-iBMCAssetTag -Session $Sessions -AssetTag $AssertTagList***   
 ***# filter all invalid result***   
 ***$InvaildSetResults = @($($SetResults | Where-Object {$\_ -is [Exception]}))***   
 ***# display result failed reason***   
 ***if ($InvaildSetResults.Count -gt 0) {***   
 ***$InvaildSetResults | ForEach-Object {***   
 ***Write-Error -Exception $\_***   
 ***}***   
 ***return***   
 ***}***   
   
 ***# get asset tag***   
 ***$GetResults = Get-iBMCAssetTag -Session $Sessions***   
 ***# filter all invalid result***   
 ***$InvaildGetResults = @($($GetResults | Where-Object {$\_ -is [Exception]}))***   
 ***# display result failed reason***   
 ***if ($InvaildGetResults.Count -gt 0) {***   
 ***$InvaildGetResults | ForEach-Object {***   
 ***Write-Error -Exception $\_***   
 ***}***   
 ***return***   
 ***}***   
 ***# output result***   
 ***$count = $iBMCAddressList.Count***   
   
 ***for($i = 0; $i -lt $count; $i++) {***   
 ***Write-Output "iBMCAddress: $($iBMCAddressList[$i])"***   
 ***Write-Output $($GetResults[$i] | fl)***   
 ***}***   
   
 # Codes end   
}   
finally {   
 # disconnect all valid sessions   
 $VaildSessions = $Sessions | Where-Object {$\_ -isnot [Exception]}   
 if ($null -ne $VaildSessions) {   
 Disconnect-iBMC -Session $VaildSessions | Out-Null   
 }   
}

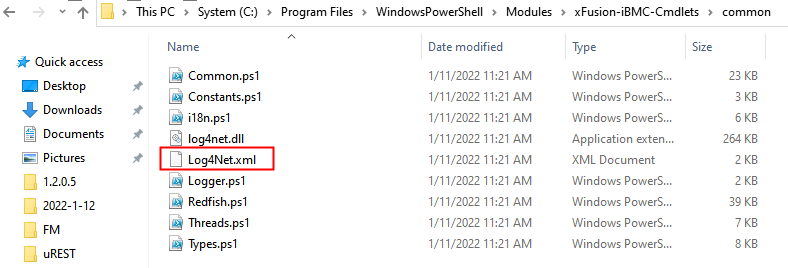
## How to Record Logs as a Non-Administrator User

Log in to the running environment of the plug-in as an administrator.

Go to the plug-in installation directory, for example, **C:\Program Files\WindowsPowerShell\Modules\xFusion-iBMC-Cmdlets\common**.

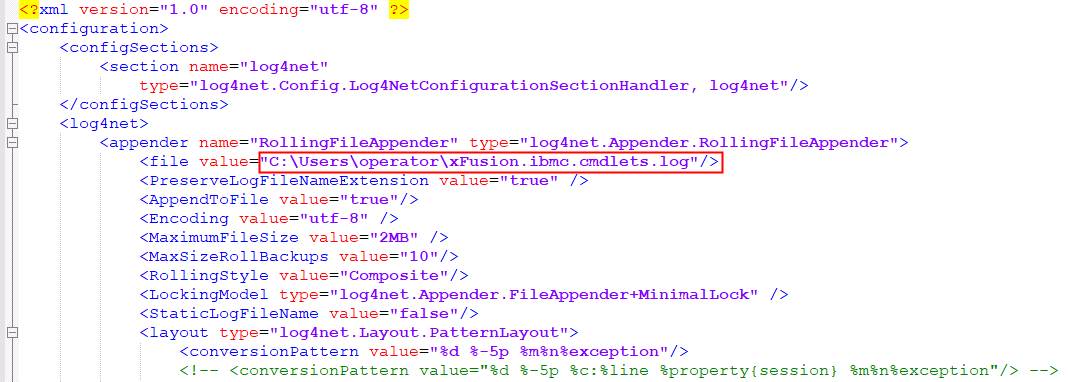
Find the **Log4Net.xml** log configuration file, as shown in Figure 4-15.

Log4Net.xml



Open the **Log4Net.xml** file in editing mode and change the log saving path to the path of a non-administrator user.

For example, in **C:\Users\operator\xfusion.ibmc.cmdlets.log**, **operator** indicates a non-administrator user.



----End

1. Public IP Addresses and Email Addresses

The *xFusion-iBMC-Cmdlets Plugin Public IP Addresses and Email Addresses* describes the public IP addresses and email addresses of the open-source and third-party software used in the xFusion-iBMC-Cmdlets plugin. For details, see the *xFusion-iBMC-Cmdlets Plugin Public IP Addresses and Email Addresses*.



You can obtain the *xFusion-iBMC-Cmdlets Plugin Public IP Addresses and Email Addresses* from the same path as the user guide.

1. Obtaining Technical Support

To obtain assistance, contact technical support as follows:

* Contact customer service center at [support@xfusion.com](mailto:support@xfusion.com)
* Contact technical support personnel.