

Using Windows® Remote Management (WinRM) to Remotely Manage Dell™ PowerEdge™ M1000e Using the Chassis Management Controller (CMC)

A Dell™ Technical White Paper

Modular Systems Management Software
By Christopher Poblete

Dell Product Group
September 2009



THIS WHITE PAPER IS FOR INFORMATIONAL PURPOSES ONLY, AND MAY CONTAIN TYPOGRAPHICAL ERRORS AND TECHNICAL INACCURACIES. THE CONTENT IS PROVIDED AS IS, WITHOUT EXPRESS OR IMPLIED WARRANTIES OF ANY KIND.

© 2009 Dell Inc. All rights reserved. Reproduction in any manner whatsoever without the express written permission of Dell, Inc. is strictly forbidden. For more information, contact Dell.

Dell, the *DELL* logo, and *PowerEdge* are trademarks of Dell Inc. *Microsoft*, *Windows*, *Windows Vista*, *Windows XP* and *Windows Server* are either trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries. Other trademarks and trade names may be used in this document to refer to either the entities claiming the marks and names or their products. Dell Inc. disclaims any proprietary interest in trademarks and trade names other than its own.

Table of Contents

Executive Summary	4
Introduction	4
Configuration	4
WinRM Client Configuration	4
SSL Server Certificate Configuration	6
New SSL Server Certificate Configuration	7
Discovery	8
WS-MAN Discovery	8
Profile Discovery	9
Blade Management	11
Performing a Blade Inventory	11
Determining Blade Power Status	12
Controlling Blade Power	13
Changing the Blade First Boot Device	15
Setting Blade Power Throttle Priority Level	16
Chassis Management	17
Changing the System Input Power Cap	17
Enabling/Disabling a Local User Account	17
Clearing the Hardware/Software Log	18
Resetting Power Statistics	18
Modifying the Power Redundancy Policy	19
Enabling/Disabling Dynamic Power Supply Engagement	19
Error Messages	20
References	21
File References	21

Executive Summary

This document provides examples for using the Windows Remote Management (WinRM) tool to access the remote management capabilities available through the Web Services for Management (WS-MAN) interface of the Chassis Management Controller (CMC). The target audience of this document is an application developer who has some WS-MAN knowledge, CIM profiles and related DMTF standard specifications in order to understand terminologies used in this document.

Introduction

The Dell™ Chassis Management Controller (CMC) is a hot-pluggable systems management hardware and software solution designed to provide remote management capabilities and power control functions for Dell PowerEdge™ M1000e chassis systems.

For information on Chassis Management Controller (CMC) refer to:

<http://support.dell.com/support/edocs/software/smdrac3/cmc/index.htm>

Web Services for Management (WS-MAN) is a Simple Object Access Protocol (SOAP)-based protocol used for systems management. WS-MAN provides an interoperable protocol for devices to share and exchange data across networks. CMC uses WS-MAN to convey Distributed Management Task Force (DMTF) Common Information Model (CIM)-based management information; the CIM information defines the semantics and information types that can be manipulated in a managed system. The Dell embedded server platform management interfaces are organized into profiles, where each profile defines the specific interfaces for a particular management domain or area of functionality. Additionally, Dell has defined a number of model and profile extensions that provide interfaces for additional capabilities.

For information on DMTF CIM Profiles refer to:

www.dmtf.org/standards/profiles

For information on Web Services for Management (WS-MAN) refer to:

www.dmtf.org/standards/wbem/wsman

For information on Windows Remote Management (WinRM) refer to:

msdn.microsoft.com/en-us/library/aa384291.aspx

Configuration

WinRM Client Configuration

Windows Vista® and Server 2008 contain a standard component call “WS-Management”. This component contains the WinRM client. For Windows XP and Server 2003, download and install this component using the link below; installation requires local administrator privilege.

[Windows WinRM Download](#)

Once installed, verify that the WinRM client is working using the identify command; for example run:

```
C: \>winrm identify
IdentifyResponse
  ProtocolVersion = http://schemas.dmtf.org/wbem/wsman/1/wsman.xsd
  ProductVendor = Microsoft Corporation
  ProductVersion = OS: 5.2.3790 SP: 2.0 Stack: 1.1
```

Next, properly configure the WinRM client to connect to CMC. Before making any changes check the existing configuration using the following command that requires the local administrator privilege:

```
C: \>winrm get winrm/config/client
Client
  NetworkDelays = 5000
  URLPrefix = wsman
  AllowUnencrypted = false
  Auth
    Basic = false
    Digest = true
    Kerberos = true
    Negotiate = true
    Certificate = true
  DefaultPorts
    HTTP = 80
    HTTPS = 443
  TrustedHosts
```

The WS-MAN interface to CMC uses the Secured Socket Layer (SSL) on port 443 to secure a connection. The connection requires you to provide credentials using the CMC local user privileges; authentication scheme support is “basic” only. If the port and authentication scheme are not set in the WinRM client, you can change the configuration with the following commands that require the local administrator privilege:

```
C: \>winrm set winrm/config/client/Auth @{Basic="true"}
Auth
  Basic = true
  Digest = true
  Kerberos = true
  Negotiate = true
  Certificate = true
```

```
C: \>winrm set winrm/config/client/DefaultPorts @{HTTPS="443"}
DefaultPorts
  HTTP = 80
  HTTPS = 443
```

The minimum configuration is set. If there are still connectivity problems, this may be attributed to local or domain network security policy. To help troubleshoot the problem, you may try the following temporary changes to the configuration that require local Administrator privilege:

```
C: \>winrm set winrm/config/client @{AllowUnencrypted="true"}
C: \>winrm set winrm/config/client @{TrustedHosts="cmc-44th123"}
```

The name “cmc-44th123” used above is a sample name and should be replaced with an actual host name. Throughout the document in the example texts, values such as the host name, user name, user password, file names and numbers should be replaced with actual values.

When making an SSL connection, WinRM by default verifies the SSL certificate name (CN) and the certificate authority (CA). The CN contains the host name of the owning system. When the IP address you provided to WinRM as the host name does not match the value in CN, the following error message is displayed:

```
WSManFault
  Message = The server certificate on the destination computer (192.168.130.157:443)
  has the following errors:
  The SSL certificate contains a common name (CN) that does not match the hostname.
```

To resolve this issue, instead of providing the IP address, provide the corresponding host name. In case the DNS is not able to resolve the host name to the IP address, you may make local changes that require the local administrator privilege to the client system by adding the host name and IP address pair to the “hosts” file located at “%SYSTEMROOT%\system32\drivers\etc\hosts”.

You may also get the following error message from WinRM if there is an issue with the server certificate:

WSManFault

Message = The server certificate on the destination computer (cmc-44TH123:443) has the following errors:
The SSL certificate is signed by an unknown certificate authority.

To resolve this issue, you need to add the CA to the local certificate store under the “Trusted Root Certificate Authority”. Refer to the next section for more info.

SSL Server Certificate Configuration

The server CA from WS-MAN should match a trusted CA in the local certificate store otherwise, the WinRM connection will fail with an error of `unknown certificate authority`. When there is no matching CA, and the Server CA from WS-MAN is trusted, you can install the server certificate to the trusted local certificate store. There are three ways to acquire the server certificate as described below.

The first method is to use the RACADM tool. Refer to the following Web site for more information:
<http://support.dell.com/support/edocs/software/smdrac3/RAC>

1. Run the `ssl certdownload` command in RACADM, for example:
`C: \>racadm -r cmc-44TH123 -u admin -p admin123 ssl certdownload -f cmc.cer -t 1`
2. Install the certificate either by double clicking the file, or by entering `start cmc.cer` from command line.
3. Carefully check all information for any errors. If there are errors, you may need to generate a new certificate See the “Making a New Server Certificate” section.
4. Click the **Install Certificate ...**, and click **Next** to continue.
5. Select **Automatically select the certificate store...**, and click **Next**.
6. Click **Finish**.

The second method is to use the OpenSSL tool. Refer to the following Web site for more information:
<http://www.openssl.org>

1. Run the `showcerts` command in OpenSSL, for example:
`C: \>openssl.exe s_client -showcerts -host cmc-44TH123 -port 443 >cmc.cer`
2. Install the certificate either by double clicking the file, or by entering `start cmc.cer` from command line.
3. Carefully check all information for any errors. If there are errors, you may need to generate a new certificate See the “Making a New Server Certificate” section.
4. Click the **Install Certificate ...**, and click **Next** to continue.
5. Select **Automatically select the certificate store...**, and click **Next**.
6. Click **Finish**.

The last method is to use the CMC Web interface.

1. Using a supported Web browser, enter the CMC Web HTTPS URL.
2. Usually the first time a browser visits an HTTPS URL it will ask to verify the server SSL certificate. If not, try clearing the history cache of the browser.
3. When the browser asks to verify the server SSL certificate, you have the option to view the certificate. Find the **View Certificate** button or link depending on the type of browser you are using.
4. The browser should give you an option to **install the certificate**; select this option to install the certificate.

5. Carefully check all information for any errors. If there are errors, you may need to generate a new certificate See the “Making a New Server Certificate” section.
6. Click the **Install Certificate ...**, and click **Next** to continue.
7. Select **Automatically select the certificate store...**, and click **Next**.
8. Click **Finish**.

New SSL Server Certificate Configuration

There may be situations where you need to create a new SSL server certificate, such as using a specific and common root CA or correcting the value of a CN. To create a new server certificate, request a CSR from the CMC Web interface using the following steps:

1. Navigate to **Network/Security** à **SSL** à **Generate a New Certificate Signing Request (CSR)** and click **Next**.
2. Provide the necessary information and click **Generate**; you will be asked to save to a local file.

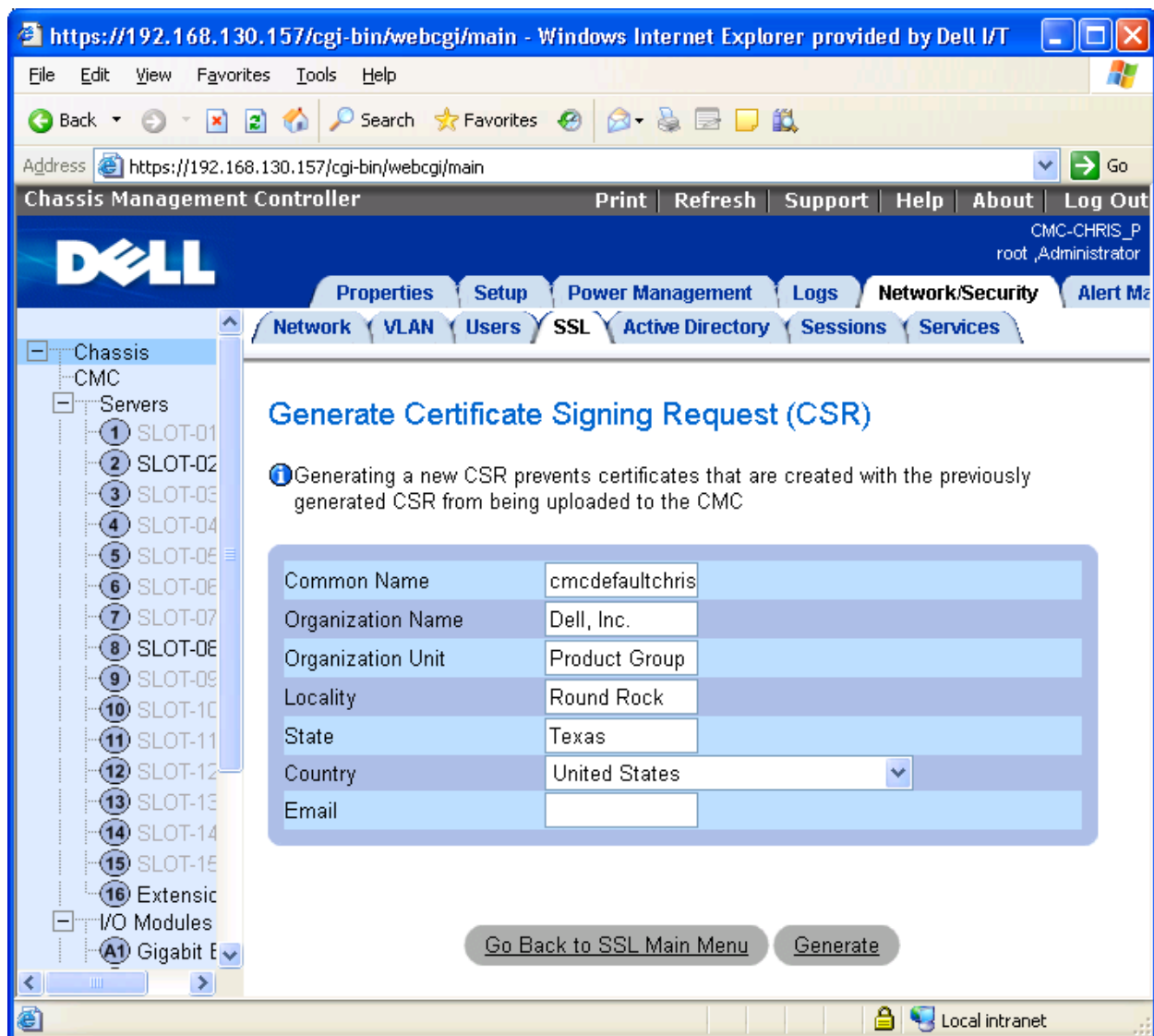


Figure 1 - Generate a CSR from the CMC GUI

3. Submit the CSR to have it signed by a Certificate Authority of your choice.

You can also locally sign the CSR using the Microsoft Windows certificate services. These services may require you to install the IIS Service to access its Web interface. Use the following basic steps to locally sign the CSR:

1. Using Internet Explorer go to <http://localhost/certsrv>
2. Click **Request a Certificate**, and then click **Advanced Certificate Request** and **Next**.
3. Select **Submit a certificate request by using a base-64-encoded CMC...**, and select **Next**.
4. Either cut-and-paste, or select **Browse for a file to insert**, the CSR generated from CMC. Submit and follow the wizard.
5. Go to **Administrative Tools -> Certification Authority**. Under **Pending Requests**, right click the CMC request and select **All Tasks** and then click **Issue**.
6. Go back to <http://localhost/certsrv>
7. Select **View the status of a pending certificate request -> Saved Request Certificate**, and select **base-64-encoded**.
8. Download the signed certificate.

Another option available to locally sign the CSR is to use the OpenSSL tool.

1. If you don't have a root CA certificate, create one using the `genrsa` and `req` commands; for example:
C: \>openssl genrsa -des3 -out test-rootca.key 1024
C: \>openssl req -new -x509 -key test-rootca.key -out test-rootca.crt -days 3650
2. Sign the CSR using the `X509` command, for example:
C: \>openssl x509 -req -in server-request.csr -out server-response.crt -sha1 -CA test-rootca.crt -CAkey test-rootca.key -CAcreateserial -days 3650

Once the CSR is signed, upload it to the CMC by navigating to **Network/Security à SSL à Upload Server Certificate** and following the wizard.

Discovery

WS-MAN Discovery

The WS-Management specification defines a simple request-response operation that is used to discover the presence of a WS-MAN service on the target system with only the transport address known. For WS-MAN discovery, use the `identify` command in WinRM.

A sample command:

```
C: \>winrm identify -u:admin -p:admin123 -r:https://cmc-44TH123 -a:basic -encoding:utf-8
```


Sample output:

```
<wsmi d: IdentifyResponse
xml ns: wsmi d="http://schemas.dmtf.org/wbem/wsman/identity/1/wsmiidentity.xsd"
xml ns: SMASH="http://schemas.dmtf.org/wbem/smash/1/smash.xsd">
  <wsmi d: ProtocolVersion>http://schemas.dmtf.org/wbem/wsman/1/wsman.xsd</wsmi d: ProtocolVersion>
  <wsmi d: ProductVendor>Dell, Inc.</wsmi d: ProductVendor>
  <wsmi d: ProductVersion>2.1</wsmi d: ProductVersion>
  <SMASH: SMASHVersion>2.0</SMASH: SMASHVersion>
  <wsmi d: SecurityProfiles>
    <wsmi d: SecurityProfileName>HTTP_TLS_1</wsmi d: SecurityProfileName>
    <wsmi d: SecurityProfileName>HTTP_TLS_2</wsmi d: SecurityProfileName>
  </wsmi d: SecurityProfiles>
</wsmi d: IdentifyResponse>
```

Profile Discovery

Features available to WS-MAN are advertised through profile registration using a common namespace. Profile registration is represented in the CIM_RegisteredProfile class, and the common namespace is root/interop. Each instance of this class provides the feature name, version of the feature specification it implements, and the implementing vendor.

A sample command to enumerate the CIM_RegisteredProfile:

```
C: \>winrm enumerate http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_RegisteredProfile?__cimnamespace=root/interop -u: admin -p: admin123 -r: https://cmc-44TH123 -a: basic -encoding: utf-8 -format: pretty
```

Sample output (partial):

```
<p: Dell_BaseServerProfile xml ns: p="http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_BaseServerProfile" xml ns: xsi="http://www.w3.org/2001/XMLSchema-instance">
  <p: InstanceID>Dell: reg5</p: InstanceID>
  <p: AdvertiseTypes>2</p: AdvertiseTypes>
  <p: RegisteredOrganization>2</p: RegisteredOrganization>
  <p: RegisteredName>Base Server</p: RegisteredName>
  <p: RegisteredVersion>1.0.0</p: RegisteredVersion>
  <p: OtherRegisteredOrganization xsi: nil="true" />
  <p: Caption xsi: nil="true" />
  <p: Description xsi: nil="true" />
  <p: ElementName xsi: nil="true" />
</p: Dell_BaseServerProfile>
```

Each instance of the CIM_RegisteredProfile class has at least one association class instance that links the profile registration to the implementation feature class. The association class also advertises the implementation namespace using the enumeration of associated classes; see the References example below).

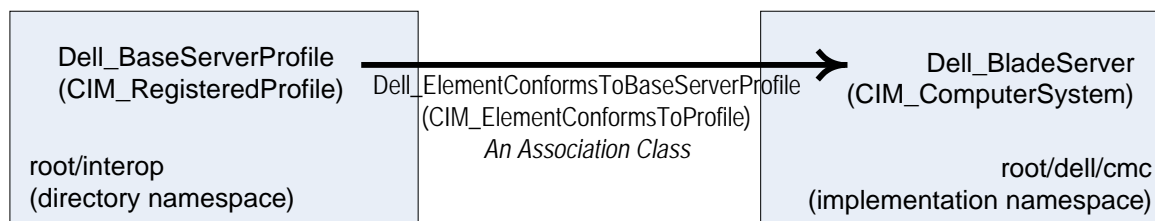


Figure 2 - Implementation Discovery

When enumerating association classes, a reference to one side of the association is required. A class reference consists of a resource URI and a set of keys; providing the reference value in a command line is challenging due to the handling of spaces within the key value. Examples of enumerating associations in this document use a VB script tool **dell-enum-ar.vbs**.

A sample command to enumerate association classes, referred to as “associators”, of the Dell_BaseServerProfile:

```
C: \>cscript /nologo dell-enum-ar.vbs associators -host: cmc-44TH123 -ns: root/interop -
user: admin -pass: admin123 -class: Dell_BaseServerProfile -
select: {InstanceID='Dell:reg5' }
```

Sample output (partial):

```
<p: Dell_ElementConformsToBaseServerProfile xmlns:p="http://schemas.dell.com/wbem/wscim/1/cim-
schema/2/Dell_ElementConformsToBaseServerProfile" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
xmlns:wsmn="http://schemas.dmtf.org/wbem/wsmn/1/wsmn.xsd">
  <p: Namespace>root/interop</p: Namespace>
  <p: ManagedElement>
    <wsa: Address>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa: Address>
    <wsa: ReferenceParameters>
      <wsmn: ResourceURI>http://schemas.dell.com/wbem/wscim/1/cim-
schema/2/Dell_BaseServerProfile</wsmn: ResourceURI>
      <wsmn: SelectorSet>
        <wsmn: Selector Name="CreationClassName">Dell_BaseServer</wsmn: Selector>
        <wsmn: Selector Name="Name">BY5TF1: blade</wsmn: Selector>
        <wsmn: Selector Name="__cimnamespace">root/dell/cmc</wsmn: Selector>
      </wsmn: SelectorSet>
    </wsa: ReferenceParameters>
  </p: ManagedElement>
  <p: ConformanceStandard>
    <wsa: Address>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa: Address>
    <wsa: ReferenceParameters>
      <wsmn: ResourceURI>http://schemas.dell.com/wbem/wscim/1/cim-
schema/2/Dell_BaseServerProfile</wsmn: ResourceURI>
      <wsmn: SelectorSet>
        <wsmn: Selector Name="InstanceID">Dell:reg5</wsmn: Selector>
        <wsmn: Selector Name="__cimnamespace">root/interop</wsmn: Selector>
      </wsmn: SelectorSet>
    </wsa: ReferenceParameters>
  </p: ConformanceStandard>
</p: Dell_ElementConformsToBaseServerProfile>
```

A sample command to enumerate associated classes, referred to as “references”, of the Dell_BaseServerProfile:

```
C: \>cscript /nologo dell-enum-ar.vbs references -host: cmc-44TH123 -ns: root/interop -
user: admin -pass: admin123 -class: Dell_BaseServerProfile -
select: {InstanceID='Dell:reg5' } -epr: 2
```

Sample output:

```
<wsmn: Item xmlns:wsmn="http://schemas.dmtf.org/wbem/wsmn/1/wsmn.xsd"
xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
xmlns="http://schemas.dmtf.org/wbem/wsmn/1/wsmn.xsd">
  <p: Dell_BaseServer xmlns:p="http://schemas.dell.com/wbem/wscim/1/cim-
schema/2/Dell_BaseServer" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cm="http://schemas.dmtf.org/wbem/wscim/1/common">
    <p: Namespace cim:Key="true">root/dell/cmc</p: Namespace>
    <p: CreationClassName cim:Key="true">Dell_BaseServer</p: CreationClassName>
    <p: Name cim:Key="true">DMCTST8: blade</p: Name>
    <p: Dedicated>0</p: Dedicated>
    <p: TransitioningToState>12</p: TransitioningToState>
    <p: EnabledDefault>2</p: EnabledDefault>
    <p: RequestedState>5</p: RequestedState>
    <p: EnabledState>3</p: EnabledState>
    <p: HealthState>5</p: HealthState>
    <p: OperationalStatus>2</p: OperationalStatus>
    <p: ElementName>Dell_BaseServer</p: ElementName>
    <p: OtherIdentifyingInfo>pkg008</p: OtherIdentifyingInfo>
    <p: OtherIdentifyingInfo>DMCTST8</p: OtherIdentifyingInfo>
    <p: IdentifyingDescriptions>CIM: Tag</p: IdentifyingDescriptions>
    <p: IdentifyingDescriptions>DCIM: ServiceTag</p: IdentifyingDescriptions>
    <p: ClassID>008</p: ClassID>
  </p: Dell_BaseServer>
  <wsa: EndpointReference>
    <wsa: Address>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa: Address>
    <wsa: ReferenceParameters>
```

```
<wsman: ResourceURI>http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_BladeServer</wsman: ResourceURI>
<wsman: SelectorSet>
  <wsman: Selector Name="__cimnamespace">root/dell/cmc</wsman: Selector>
  <wsman: Selector Name="Name">DMCTST8: blade</wsman: Selector>
  <wsman: Selector Name="CreationClassName">Dell_BladeServer</wsman: Selector>
</wsman: SelectorSet>
</wsa: ReferenceParameters>
</wsa: EndpointReference>
</wsman: Item>
```

In the response XML above, the associated reference class of “Dell_BaseServerProfile” in the root/interop namespace is a “Dell_BladeServer” class in the root/dell/cmc namespace. The namespace is found in the following node: EndpointReference à ReferenceParameters à SelectorSet à Selector à __cimnamespace.

Blade Management

Performing a Blade Inventory

To get an inventory of blade servers that are present in my chassis enclosure, either enumerate the Dell_BladeServer class or enumerate the CIM_ComputerSystem class with filter of “where CreationClassName=’Dell_BladeServer’”. A sample output is provided in the previous section. An instance of this class represents the presence of a blade server in the chassis. The other significant information listed is the “service tag” associated with the blade server.

Other blade server inventory information is available in the associated classes. Each instance of Dell_BladeServer has an association to the class Dell_BladePackage through an association class Dell_BladeSysPkg. Dell_BladePackage that contains the blade’s model number, serial number, and part number. Also, the blade slot number location in the chassis is available. This information is provided in the Dell_BladeSlot class. Each instance of Dell_BladePackage has an association to the class Dell_BladeSlot through an association class Dell_BladePkgConn.

A sample command to enumerate references of Dell_BladeServer and to find Dell_BladePackage:

```
C:\>cscript /nologo dell-enum-ar.vbs references -host: cmc-44TH123 -user: admin -
pass: admin123 -class: Dell_BladeServer -
select: {Name='DMCTST8: blade'; CreationClassName='Dell_BladeServer' }
```

Sample output filtered for Dell_BladePackage:

```
<p: Dell_BladePackage xml ns: p="http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_BladePackage" xml ns: xsi="http://www.w3.org/2001/XMLSchema-instance"
xml ns: cim="http://schemas.dmtf.org/wbem/wscim/1/common">
  <p: Namespace>root/dell/cmc</p: Namespace>
  <p: Tag cim: Key="true">pkg008</p: Tag>
  <p: CreationClassName cim: Key="true">Dell_BladePackage</p: CreationClassName>
  <p: VendorCompatibilityStrings>Dell: Blade: PowerEdgeM1000e: Dell_BladePackage</p: VendorCompatibilityStrings>
  <p: PackageType>16</p: PackageType>
  <p: RemovalConditions>2</p: RemovalConditions>
  <p: Name>PowerEdgeM805</p: Name>
  <p: ElementName>Dell_Blade_Server</p: ElementName>
  <p: Manufacturer>70163</p: Manufacturer>
  <p: Model>PowerEdgeM805</p: Model>
  <p: SerialNumber>CN7016386N006M</p: SerialNumber>
  <p: PartNumber>OD413F</p: PartNumber>
  <p: CanBeFRUed>true</p: CanBeFRUed>
  <p: ClassId>008</p: ClassId>
</p: Dell_BladePackage>
```

A sample command to enumerate references of Dell_BladePackage and to find Dell_BladeSlot:

```
C:\>cscript /nologo dell-enum-ar.vbs references -host: cmc-44TH123 -user: admin -
pass: admin123 -class: Dell_BladePackage -
select: {Tag=' pkg008' ; CreationClassName=' Dell_BladePackage' }
```

Sample output filtered for Dell_BladeSlot:

```
<p: Dell_BladeSlot xmlns:p="http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_BladeSlot"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:cim="http://schemas.dmtf.org/wbem/wscim/1/common">
  <p: Namespace>root/dell/cmc</p: Namespace>
  <p: CreationClassName cim:Key="true">Dell_BladeSlot</p: CreationClassName>
  <p: Tag cim:Key="true">slot008</p: Tag>
</p: Dell_BladeSlot>
<p: VendorCompatibilityStrings>Dell: Blade: PowerEdgeM1000e: Dell_BladeSlot</p: VendorCompatibilityStrings>
  <p: Number>8</p: Number>
  <p: Manufacturer>Dell, Inc.</p: Manufacturer>
  <p: ElementName>Blade Slot</p: ElementName>
  <p: ConnectorLayout>7</p: ConnectorLayout>
  <p: ClassId>008</p: ClassId>
</p: Dell_BladeSlot>
```

Determining Blade Power Status

To determine a blade server power status, either on or off, use the association class CIM_AssociatedPowerManagementService (APMS). For a given Dell_BladeServer, enumerate its associators and look for an instance of Dell_BladeAssocPwrSvc that is a derivation of APMS. The power status is represented in the "PowerState" property. The value mapping is as follows:

PowerState possible values:

- 1 – Powering up or Unknown
- 2 – Power is on
- 6 – Power is off

A sample command to enumerate associators of Dell_BladeServer and find Dell_BladeAssocPwrSvc:

```
C:\>cscript /nologo dell-enum-ar.vbs associators -host: cmc-44TH123 -user: admin -
pass: admin123 -class: Dell_BladeServer -
select: {CreationClassName=' Dell_BladeServer' ; Name=' DMCTST8: blade' }
```

Sample output filtered for Dell_BladeAssocPwrSvc:

```
<p: Dell_BladeAssocPwrSvc xmlns:p="http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_BladeAssocPwrSvc" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
xmlns:wsman="http://schemas.dmtf.org/wbem/wsman/1/wsman.xsd">
  <p: Namespace>root/dell/cmc</p: Namespace>
  <p: UserOfService>
    <wsa: Address>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa: Address>
    <wsa: ReferenceParameters>
      <wsman: ResourceURI>http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_BladeServer</wsman: ResourceURI>
      <wsman: SelectorSet>
        <wsman: Selector Name="CreationClassName">Dell_BladeServer</wsman: Selector>
        <wsman: Selector Name="Name">DMCTST8: blade</wsman: Selector>
      </wsman: SelectorSet>
    </wsa: ReferenceParameters>
  </p: UserOfService>
  <p: ServiceProvided>
    <wsa: Address>http://schemas.xmlsoap.org/ws/2004/08/addressing/role/anonymous</wsa: Address>
    <wsa: ReferenceParameters>
      <wsman: ResourceURI>http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_PowerMgmtSvc</wsman: ResourceURI>
      <wsman: SelectorSet>
        <wsman: Selector Name="SystemCreationClassName">Dell_ComputerSystem</wsman: Selector>
        <wsman: Selector Name="SystemName">1: Modular</wsman: Selector>
      </wsman: SelectorSet>
    </wsa: ReferenceParameters>
  </p: ServiceProvided>
```

```
<wsman: Selector Name="CreationClassName">Dell_PowerMgmtSvc</wsman: Selector>
<wsman: Selector Name="Name">pwrSvc1</wsman: Selector>
</wsman: SelectorSet>
</wsa: ReferenceParameters>
</p: ServiceProvided>
<p: PowerState>2</p: PowerState>
<p: OtherPowerState xsi:nil="true"/>
<p: RequestedPowerState>1</p: RequestedPowerState>
<p: OtherRequestedPowerState>No Change</p: OtherRequestedPowerState>
<p: PowerOnTime xsi:nil="true"/>
</p: Dell_BladeAssocPwrSvc>
```

Controlling Blade Power

Controlling the power of a blade server is provided by the class CIM_PowerManagementService (PMS). The control capabilities are provided by the class CIM_PowerManagementCapabilities (PMC). There are two properties in PMC that describe the power control capabilities described as follows:

PowerStateSupported values:

- 2 – Power On is supported
- 5 – Power Cycle is supported
- 8 – Power Off is supported

PowerChangeCapabilities values:

- 3 – Power control is supported
- 4 – Power cycling is supported

Once capabilities are determined, the power control is performed by invoking a method in the PMS class; invoking a method requires you to provide the instance of the class. The power control method "RequestPowerStateChange" requires 2 input parameters; one is the instance of the Dell_BladeServer to which the power control is to be applied and the other is the power state action to be performed.

Method specification:

Class name	Dell_PowerMgmtSvc	This is a single instance
Method name	RequestPowerStateChange	
Parameter 1	PowerState	uint16 possible values: 2 = Power on 5 = Power cycle (cold boot) 8 = Power off 32768 = Reset (warm boot) 32769 = Gracefully power off
Parameter 2	ManagedElement	Reference of Dell_BladeServer

A sample command to enumerate PMC:

```
C: \>winrm enumerate http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_PowerManagementCapabilities?__cimnamespace=root/dell/cmc -u: admin -p: admin123 -r: https://cmc-44TH123 -a: basic -encoding: utf-8 -options: {ShowExtensions="true"} -format: pretty
```

Sample output:

```
<p: Dell_PowerMgmtCap xmlns:p="http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_PowerMgmtCap" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:cim="http://schemas.dmtf.org/wbem/wscim/1/common">
<p: InstanceID cim:Key="true">Dell: pscap1</p: InstanceID>
<p: ElementName>Power Management Capabilities</p: ElementName>
<p: PowerStatesSupported>2</p: PowerStatesSupported>
<p: PowerStatesSupported>5</p: PowerStatesSupported>
<p: PowerStatesSupported>8</p: PowerStatesSupported>
<p: PowerChangeCapabilities>3</p: PowerChangeCapabilities>
```

```
<p: PowerChangeCapabilities>4</p: PowerChangeCapabilities>
</p: Dell_PowerMgmtCap>
```

A sample command to enumerate PMS:

```
C: \>winrm enumerate http://schemas.dmtf.org/wbem/wscim/1/cim-schema/2/CIM_PowerManagementService?__cimnamespace=root/dell/cmc -u: admin -p: admin123 -r: https://cmc-44TH123 -a: basic -encoding: utf-8 -options: {ShowExtensions="true"} -format: pretty
```

Sample output:

```
<p: Dell_PowerMgmtSvc xmlns:p="http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_PowerMgmtSvc" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:cim="http://schemas.dmtf.org/wbem/wscim/1/common">
  <p: Name cim:Key="true">pwrSvc1</p: Name>
  <p: CreationClassName cim:Key="true">Dell_PowerMgmtSvc</p: CreationClassName>
  <p: SystemName cim:Key="true">1: Modular</p: SystemName>
  <p: SystemCreationClassName cim:Key="true">Dell_ComputerSystem</p: SystemCreationClassName>
  <p: ElementName>Power Management Service</p: ElementName>
  <p: EnabledState>5</p: EnabledState>
  <p: RequestedState>12</p: RequestedState>
  <p: EnabledDefault>2</p: EnabledDefault>
</p: Dell_PowerMgmtSvc>
```

A sample command to invoke RequestPowerStateChange using the instance keys in the output above:

```
C: \>winrm invoke RequestPowerStateChange http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_PowerMgmtSvc?__cimnamespace=root/dell/cmc+Name=pwrSvc1+CreationClassName=Dell_PowerMgmtSvc+SystemName=1: Modular+SystemCreationClassName=Dell_ComputerSystem -r: https://cmc-44TH123 -u: admin -p: admin123 -a: basic -encoding: utf-8 -file: input-RequestPowerStateChange.xml
```

A sample input XML to describe the method input parameters:

```
<n1: RequestPowerStateChange_INPUT xmlns:n1="http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_PowerMgmtSvc" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing" xmlns:wsman="http://schemas.dmtf.org/wbem/wsman/1/wsman.xsd">
  <n1: PowerState>5</n1: PowerState>
  <n1: ManagedElement>
    <wsa: ReferenceParameters>
      <wsman: ResourceURI>http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_BladeServer</wsman: ResourceURI>
      <wsman: SelectorSet>
        <wsman: Selector Name='__cimnamespace'>root/dell/cmc</wsman: Selector>
        <wsman: Selector Name='CreationClassName'>Dell_BladeServer</wsman: Selector>
        <wsman: Selector Name='Name'>5YYYSF1: blade</wsman: Selector>
      </wsman: SelectorSet>
    </wsa: ReferenceParameters>
  </n1: ManagedElement>
</n1: RequestPowerStateChange_INPUT>
```

Sample response:

```
<n1: RequestPowerStateChange_OUTPUT xmlns:n1="http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_PowerMgmtSvc">
  <n1: ReturnValue>0</n1: ReturnValue>
</n1: RequestPowerStateChange_OUTPUT>
```

An alternative to using the PMS class for power control actions is to use the “RequestStateChange” method of the class Dell_BladeServer. This method requires an instance of Dell_BladeServer, and one input parameter “RequestedChange”.

Method specification:

Class name	Dell_BladeServer	One instance for each blade present
Method name	RequestStateChange	
Parameter 1	RequestedChange	uint16 possible values:

		2 = Power on 11 = Power cycle (cold boot) 3 = Power off 32768 = Reset (warm boot) 32769 = Gracefully power off
--	--	--

Changing the Blade First Boot Device

There are two properties that you can change in the first boot device feature; the first property you can change is the “Boot Device”. The Boot Device property can be changed using the method “ChangeBootOrder” in the class CIM_BootConfigSetting. The other property you can change is the “Boot Once” feature; this can be changed using the method “SetBootConfigRole” in the class CIM_BootService.

The “First Boot Device” setting overrides the normal BIOS setting for a blade server. If you set the first boot device to boot only once, this setting will automatically revert to the normal BIOS setting at the next boot cycle.

Method specification:

Class name	Dell_BootService	This is a single instance
Method name	SetBootConfigRole	
Parameter 1	Role	uint16 possible values: 0 = Is Next 1 = Boot Once 3 = Is Not Next The value “3” corresponds to “None” as is used in the CMC GUI and CLI.
Parameter 2	BootConfigSetting	Reference of Dell_BootConfigSetting

A sample command:

```
C: \>winrm invoke SetBootConfigRole http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_BootService?__cimnamespace=root/dell/cmc+Name=Dell:bootsvc1+SystemName=chassismgr1+SystemCreationClassName=Dell_ChassisMgr+CreationClassName=Dell_BootService -r: https://cmc-44TH123 -u: admin -p: admin123 -a: basic -encoding: utf-8 -file: input-SetBootConfigRole.xml
```

A sample input XML file that applies “Is Next” to the blade server 2 configuration:

```
<n1: SetBootConfigRole_INPUT xmlns:n1="http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_BootService" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing" xmlns:wsman="http://schemas.dmtf.org/wbem/wsman/1/wsman.xsd">
  <n1: Role>0</n1: Role>
  <n1: BootConfigSetting>
    <wsa: ReferenceParameters>
      <wsman: ResourceURI>http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_BootConfigSetting</wsman: ResourceURI>
      <wsman: SelectorSet>
        <wsman: Selector Name='__cimnamespace'>root/dell/cmc</wsman: Selector>
        <wsman: Selector Name='InstanceID'>Dell:bootcfgset02</wsman: Selector>
      </wsman: SelectorSet>
    </wsa: ReferenceParameters>
  </n1: BootConfigSetting>
</n1: SetBootConfigRole_INPUT>
```

Method specification:

Class name	Dell_BootConfigSetting	One instance for each blade present
Method name	ChangeBootOrder	

Parameter 1	Source	<p>Dell_BootSourceSetting possible values:</p> <ul style="list-style-type: none"> • pxe (PXE) • hd (Hard drive) • cd (Local CD/DVD) • vfloppy (Virtual floppy) • vcd (Virtual CD/DVD) • iscsi (iSCSI) • sd (Local SD card) • floppy (Local floppy) <p>To set the first boot device to “None”, Use the “SetBootConfigRole” method and select “3” for Role.</p>
-------------	--------	---

A sample command:

```
C:\>winrm invoke ChangeBootOrder http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_BootConfigSetting?__cimnamespace=root/dell/cmc+InstanceID=Dell:bootcfgset02 -r: https://cmc-44TH123 -u: admin -p: admin123 -a: basic -encoding: utf-8 -file: input-ChangeBootOrder.xml
```

A sample input XML file that applies “Virtual Floppy” as the first boot device:

```
<n1:ChangeBootOrder_INPUT xmlns:n1="http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_BootConfigSetting" xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing" xmlns:wsmn="http://schemas.dmtf.org/wbem/wsmn/1/wsmn.xsd">
  <n1:Source>
    <wsa:ReferenceParameters>
      <wsmn:ResourceURI>http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_BootSourceSetting</wsmn:ResourceURI>
      <wsmn:SelectorSet>
        <wsmn:Selector Name="InstanceID">Dell:bootset:vfloppy</wsmn:Selector>
      </wsmn:SelectorSet>
    </wsa:ReferenceParameters>
  </n1:Source>
</n1:ChangeBootOrder_INPUT>
```

Setting Blade Power Throttle Priority Level

When the maximum power limit for the chassis enclosure is exceeded, the servers are throttled based on the set priority level. The priority level is modeled as “Weights” in the class CIM_ResourceAllocationSettingData; set the Weight property to change a blade’s priority level value.

Set specification:

Class name	Dell_BIPwrCurResourceAllocationSettingData Dell_BIPwrResourceAllocationSettingData	Single instance
Property 1	Weight	uint64 possible values: (1 ... 9)

A sample command to set the priority level to 5:

```
C:\>winrm set http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_BIPwrCurResourceAllocationSettingData?__cimnamespace=root/dell/cmc+InstanceID=Dell:Dell_BIPwrCurResourceAllocationSettingData:2 -r: https://cmc-44TH123 -a: basic -u: admin -p: admin123 @{Weight="5"}
```


Chassis Management

Changing the System Input Power Cap

The system input power cap, or user budget, is the maximum AC power that the system is allowed to allocate to the servers and chassis infrastructure. The user budget is modeled as “Reservation” in the class CIM_ResourceAllocationSettingData; set the Reservation property to change the value of the user budget.

Set specification:

Class name	Dell_ChPwrResSettingData	Single instance
Property 1	Reservation	uint64 possible values: (refer to CMC User's Guide)

A sample command to view the current user budget setting:

```
C: \>winrm enumerate http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_ChPwrResSettingData?__cimnamespace=root/dell/cmc -u: admin -p: admin123 -r: https://cmc-44TH123 -a: basic -encoding: utf-8 -format: pretty
```

Sample output:

```
<wsman: Results xmlns:wsman="http://schemas.dmtf.org/wbem/wsman/1/wsman/results">
  <p: Dell_ChPwrResSettingData xmlns:p="http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_ChPwrResSettingData" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <p: InstanceID>Dell: chassispwr</p: InstanceID>
    <p: ChangeableType>2</p: ChangeableType>
    <p: ResourceType>28</p: ResourceType>
    <p: PoolID>Dell: ChassisPowerPool1</p: PoolID>
    <p: AllocationUnits>watts</p: AllocationUnits>
    <p: Reservation>7500</p: Reservation>
    <p: Weight>0</p: Weight>
    <p: ClassIDNum>20</p: ClassIDNum>
  </p: Dell_ChPwrResSettingData>
</wsman: Results>
```

A sample command to change the user budget value:

```
C: \>winrm set http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_ChPwrResSettingData?__cimnamespace=root/dell/cmc+InstanceID=Dell: chassispwr @{Reservation="6000"} -u: admin -p: admin123 -r: https://cmc-44TH123 -a: basic -encoding: utf-8 -format: pretty
```

Sample response:

```
<p: Dell_ChPwrResSettingData xmlns:p="http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_ChPwrResSettingData" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <p: InstanceID>Dell: chassispwr</p: InstanceID>
  <p: Weight>0</p: Weight>
  <p: Reservation>6000</p: Reservation>
  <p: AllocationUnits>watts</p: AllocationUnits>
  <p: PoolID>Dell: ChassisPowerPool1</p: PoolID>
  <p: ResourceType>28</p: ResourceType>
  <p: ChangeableType>2</p: ChangeableType>
</p: Dell_ChPwrResSettingData>
```

Enabling/Disabling a Local User Account

Using the RequestStateChange method of an instance of class CIM_Account, a user's access to the CMC can be enabled/granted or disabled/blocked; an instance of CIM_Account indicates that the user account exists. Enumerate CIM_Account or Dell_Account to determine the account you wish to change.

Method specification:

Class name	Dell_Account	One instance for each created account
Method name	RequestStateChange	
Parameter 1	RequestedState	uint16 possible values: 2 = Enable 3 = Disable

A sample command to disable the second user account:

```
C: \>winrm invoke RequestStateChange http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_Account?__cimnamespace=root/dell/cmc+Name=account2+SystemName=chassismgr1+SystemCreationClassName=Dell_ChassisMgr+CreationClassName=Dell_Account -r: https://cmc-44TH123 -u: admin -p: admin123 -a: basic -encoding: utf-8 @{RequestedState="3"}
```

Clearing the Hardware/Software Log

Using the ClearLog method of an instance of class CIM_Log, all entries of either the hardware log or software (CMC) log can be cleared or deleted. A new log entry is created to indicate that a clear log action was performed.

Method specification:

Class name	Dell_SWLog Dell_HWLog	Single instance
Method name	ClearLog	

A sample command to clear the software log:

```
C: \>winrm invoke ClearLog http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_HWLog?__cimnamespace=root/dell/cmc+InstanceId=Dell:HardwareLog -r: https://cmc-44TH123 -u: admin -p: admin123 -a: basic -encoding: utf-8
```

Resetting Power Statistics

Using the ControlMetrics method of an instance of class CIM_PowerMetricService, the “peak and minimum system power consumption statistics” can be reset. The statistics refer to the information reported under the **Real-Time Power Statistics** window of the CMC user interface.

Method Specification:

Class name	Dell_PowerMetricService	Single instance
Method name	ControlMetrics	
Parameter 1	MetricCollectionEnabled	uint16 possible values: 4 = Reset

A sample command to reset the statistics:

```
C: \>winrm invoke ControlMetrics http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_PowerMetricService?__cimnamespace=root/dell/cmc+Name=MServicel+CreationClassName=Dell_PowerMetricService+SystemName=1:Modular+SystemCreationClassName=Dell_ComputerSystem -r: https://cmc-44TH123 -u: admin -p: admin123 -a: basic -encoding: utf-8 @{MetricCollectionEnabled="4"}
```

Modifying the Power Redundancy Policy

The chassis power redundancy policy can be modified using the AssignPowerRedundancyPriority method of an instance of class CIM_PowerConfigService. The policy can be modified to one of the following values: No Redundancy; Power Supply Redundancy; or AC Redundancy.

Method specification:

Class name	Dell_PowerConfigSvc	Single instance
Method name	AssignPowerRedundancyPriority	
Parameter 1	RequestedRedundancyPriority	uint16 possible values: 2 = No Redundancy 3 = Power Supply Redundancy 4 = AC Redundancy

A sample command to modify the power supply redundancy policy to Power Supply Redundancy:

```
C: \>winrm invoke AssignPowerRedundancyPriority
http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_PowerConfigSvc?__cimnamespace=root/dell/cmc+Name=PowerConfigurati onService1+Creati onClassName=Dell_PowerConfigSvc+SystemName=chassismgr1+SystemCreati onClassName=Dell_ChassisMgr -r: https://cmc-44TH123 -u: admin -p: admin123 -a: basic -encoding: utf-8 @{RequestedRedundancyPriority="3"}
```

Related Method specification:

Class name	Dell_PowerConfigSvc	Single instance
Method name	ConfigureExternalPowerDomainDefaults	
Parameter 1	OwningComputerSystem	Reference of Dell_Modular
Parameter 2	RequestedChange	uint16 possible values: 2 = AC Redundancy 3 = Power Supply Redundancy

Enabling/Disabling Dynamic Power Supply Engagement

When the **Dynamic Power Supply Engagement** feature is enabled, the power supplies are turned either “on” or “off” based on the power consumption, thereby optimizing the energy consumption of the entire chassis. This feature is modeled as “DynamicEngageEnabled” in the class CIM_PowerConfigurationService; set DynamicEngageEnabled to enable or disable this feature.

Set specification:

Class name	Dell_PowerConfigSvc	Single instance
Property 1	DynamicEngageEnabled	uint64 possible values: 2 = Enable 3 = Disable

A sample command to set **Dynamic Engagement** to enable:

```
C: \>winrm set http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_PowerConfigSvc?__cimnamespace=root/dell/cmc+SystemCreati onClassName=Dell_ChassisMgr+SystemName=chassismgr1+Creati onClassName=Dell_PowerConfigSvc+Name=PowerConfi gurati onService1 -r: https://cmc-44TH123 -a: basic -u: admin -p: admin123
@{DynamicEngageEnabled="2"}
```

Error Messages

The CIM method invocation response may include information that describes a problem when an error occurs. Error information has three parts:

- The first part is a “MessageID” that uniquely identifies an error message format.
- The second part is a series of “MessageArguments” that contain parts of the error message.
- The last part is the “Message” which contains the error message itself.

The error message contained in the “Message” is in English; the purpose of the first two parts is to enable translation. To translate an error message into a different language, use the table below that matches the MessageID to a message template. The template consists of texts and substitution variables; refer to [Format Specification](#). The text in the template can be translated into different languages; whereas the text to be substituted in the template comes from the MessageArguments in sequence and is not to be translated.

MessageID	Message Template
CMC0001	%s: Parameter: %s not found, CMGetArg returned: %s
CMC0002	%s: Parameter: %s must be type: %s found: %s
CMC0003	%s: Parameter: %s must be type: %s found type: %hu
CMC0004	%s: Parameter: %s value not supported, found: %hu
CMC0005	%s: Get instance of parameter: %s failed, status: %s
CMC0006	%s: Property: %s of parameter: %s not found, status: %s
CMC0007	%s: Property: %s of parameter: %s must be numeric, found: %s
CMC0008	%s: Get instance of class: %s failed, status: %s
CMC0009	%s: Property: %s of class: %s not found, status: %s
CMC000A	%s: Invoke method: %s failed, status: 0x%X
CMC000B	%s: Property: %s of class: %s must be numeric, found: %s
CMC000C	%s: Parameter: %s must be reference of class: %s
CMC000D	%s: Parameter: %s at index: %u must be type: %s found type: %hu
CMC000E	%s: Parameter: %s at index: %u not found, CMGetArrayElementAt returned: %u
CMC000F	%s: Get instance of parameter: %s at index: %u failed, status: %s
CMC0010	%s: Parameter: %s at index: %u must be reference of class: %s
CMC0011	%s: Parameter: %s as reference of class: %s not found
CMC0012	%s: Count of parameter: %s must match count of parameter: %s
CMC0013	%s: Parameter: %s of type: %s must be one of: %s

A sample error message:

```
<n1: AssignPowerRedundancyPriority_OUTPUT xmlns:n1="http://schemas.dell.com/wbem/wscim/1/cim-schema/2/Dell_PowerConfigSvc">
  <n1: ReturnValue>4</n1: ReturnValue>
  <n1: MessageID>CMC000A</n1: MessageID>
  <n1: Message>AssignPowerRedundancyPriority_PowerConfigSvc: Invoke method: psu_redundancy_set_AC failed, status: 0x550F</n1: Message>
  <n1: MessageArguments>AssignPowerRedundancyPriority_PowerConfigSvc</n1: MessageArguments>
  <n1: MessageArguments>psu_redundancy_set_AC</n1: MessageArguments>
  <n1: MessageArguments>0x550F</n1: MessageArguments>
</n1: AssignPowerRedundancyPriority_OUTPUT>
```

References

- [Distributed Management Task Force \(DMTF\)](#)
- [About Windows Remote Management \(WinRM\)](#)
- [Dell™ Chassis Management Controller \(CMC\)](#)
- [OpenSSL SSL Certificate Toolkit](#)
- [Windows WinRM Download](#)

File References



dell-enum-ar.vbs