An Overview of New Features in the Dell EMC iDRAC9 3.30.30.30 RESTful API

Abstract

This Dell EMC Technical White Paper provides an overview of enhancements to the iDRAC9 RESTful API included in firmware version 3.30.30.30. The new APIs are illustrated with Python scripts.

March 2019

Revisions

Date	Description
March 2019	Initial release

Acknowledgements

This technical white paper was produced by the following members of the Dell EMC product group team:

Authors: **Paul Rubin**—Senior Product Manager, Dell EMC Server Solutions
Texas Roemer—Senior Test Engineer, Dell EMC Server Solutions
Hari Venkatachalam—Software Principal Engineer, Dell EMC Server Solutions
Srinivasulu E—Senior Software Engineer, Dell EMC Server Solutions
Chinmay Hegde—Firmware Engineer, Dell EMC Server Solutions

Support—Sheshadri PR Rao (InfoDev)

Contents

Executive summary

iDRAC with Lifecycle Controller provides a range of standards-based Applications Programming Interfaces (APIs) that enable scalable and automated management of Dell EMC PowerEdge servers. Standard Systems Management APIs have been developed by organizations such as the Institute of Electrical and Electronics Engineers (IEEE) and Distributed Management Task Force (DMTF). These APIs are widely used by commercial Systems Management products and by custom programs and scripts developed by IT staff to automate management functions such as discovery, inventory, health status checking, configuration, update, and power management.

The iDRAC RESTful API provides a modern, secure, and scalable interface for PowerEdge server management, supporting both the DMTF Redfish standard and value-added Dell EMC operations.

This technical whitepaper provides an overview of iDRAC RESTful API enhancements provided with iDRAC9 firmware version 3.30.30.30 and later, illustrating the new APIs with scripted examples.

1 Integrated Dell Remote Access Controller RESTful API

The integrated Dell remote access controller (iDRAC) delivers advanced, agent-free, local and remote server administration. Embedded in every Dell EMC PowerEdge 14th Generation (14G) server, iDRAC9 provides a secure way to automate a multitude of common management tasks. Because iDRAC is embedded, there is no additional software to install—just plug in power and network cables, and iDRAC9 is ready to go. Even before installing an Operating System (OS) or hypervisor, IT administrators have a complete set of server management features at their fingertips.

With iDRAC9 in place across the PowerEdge portfolio, the same IT administration techniques and tools can be applied throughout. This consistent management capability enables easy scaling of PowerEdge servers as an organization's infrastructure grows.

iDRAC9 provides multiple interfaces for human and automated operations including an HTML5-based Graphical User Interface (GUI), the RACADM Command Line Interface (CLI), and a range of standards-based APIs including Redfish, WS-Man, IPMI, and SNMP. These interfaces provide full lifecycle server management including inventory, configuration, monitoring, firmware update, and server retirement.

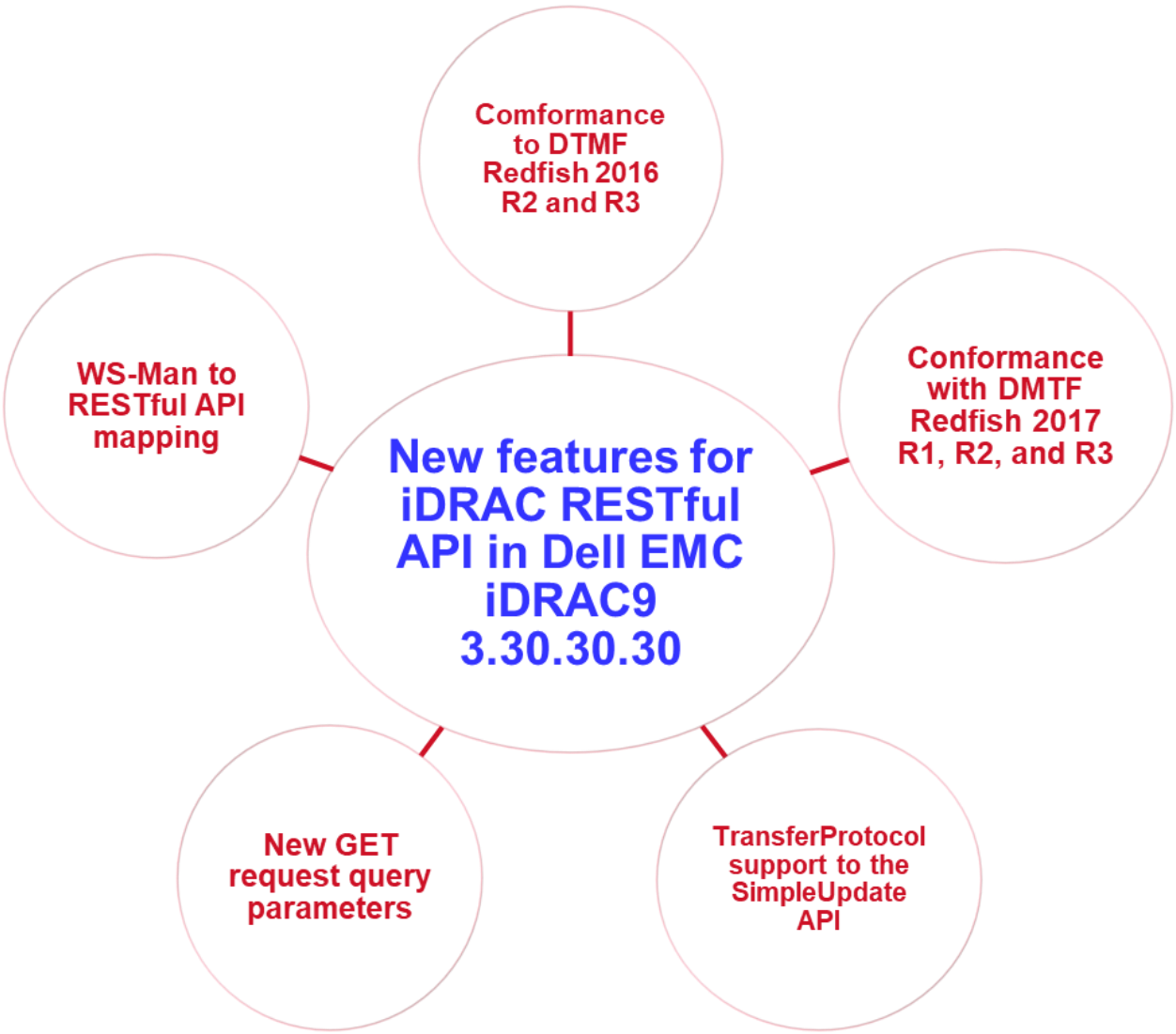
The iDRAC RESTful API provides a modern, secure, and scalable interface for PowerEdge server management, supporting both the DMTF Redfish standard and value-added Dell EMC operations. Redfish is an open industry-standard specification and schema designed to meet the requirements of IT administrators for simple, modern, and secure management of scalable platform hardware. Dell EMC is a key contributor to the iDRAC RESTful API standard, acting as co-chair of the DMTF Redfish Forum, promoting the benefits of iDRAC RESTful API, and working to deliver those benefits within industry-leading systems management solutions. iDRAC RESTful API and Redfish use a data model representation inside a hypermedia RESTful interface. The data model is defined in terms of a standard, machine-readable schema, with the payload of the messages expressed in JSON and the OData v4 protocol. Dell EMC has been supporting the Redfish standard within the iDRAC RESTful API since 2016.

1.1 What's new for iDRAC RESTful API in iDRAC9 3.30.30.30

iDRAC9 firmware version 3.30.30.30 expands the range of standards-based server management functions available through Redfish while also providing a powerful transition tool for PowerEdge customers wishing to update their automation from the SOAP–based WS-Man API to Redfish. Among the iDRAC RESTful API updates provided by firmware 3.30.30.30 are:

- Conformance with DMTF Redfish 2016 R2 and R3
  - PCIe device inventory, including non-Dell EMC devices
  - NVDIMM inventory and metrics
  - Privilege registry defining iDRAC user authorizations
  - Host interface API enabling future OS access to Redfish
- Conformance with DMTF Redfish 2017 R1, R2, and R3
  - Boot source configuration
  - Virtual media insert or eject
  - Maintenance window API to schedule configuration changes and firmware updates
- Addition of TransferProtocol support to the SimpleUpdate API
- Support for new GET request query parameters
- WS-Man to RESTful API mapping:

- Phase I—a Dell EMC OEM extension to Redfish which enables Redfish-like access to PowerEdge WS-Man attributes and operations.
- Phase II—Available in a future release, will add support for additional WS-Man features.



2 iDRAC RESTful API Redfish enhancements

The iDRAC RESTful API continues to expand the range of standards-based server management for PowerEdge servers with the release of iDRAC9 3.30.30.30 version. The following sections of the whitepaper describe the new APIs and illustrate their use with Python scripting examples—These examples are available from the iDRAC RESTful API GitHub repository https://github.com/dell/iDRAC-Redfish-Scripting.

2.1 Manage virtual media

iDRAC9 virtual media is a powerful tool supporting server provisioning and update. Virtual media enables administrators to attach image files that can be booted and accessed as if they are physical DVDs or CDs connected to the server. The virtual media management APIs enable IT developers to mount image files located on HTTP and HTTPS network shares, to check the status of mounted images, and to unmount images after use.

The script example here uses the Python script `InsertEjectVirtualMediaREDFISH.py` to perform the virtual media operations:

1. If executing the script for the first time, its recommended to view the help text:

```
C:\>InsertEjectVirtualMediaREDFISH.py -h
usage: InsertEjectVirtualMediaREDFISH.py [-h] -ip IP -u U -p P [-c C] [-o O]
[-d D] [-i I]
Python script using Redfish API to either get virtual media information,
insert or eject virtual media
```

Positional arguments:

```
InsertEjectVirtualMediaREDFISH.py -ip 192.168.0.120 -u root -p calvin -c y
```

this example will get the status of virtual media.

```
InsertEjectVirtualMediaREDFISH.py -ip 192.168.0.120 -u root -p calvin -o 1 -
                          d 1 -i http://192.168.0.130/esxi_5u1.iso
```

this example shows booting to CD ISO on HTTP share.

```
InsertEjectVirtualMediaREDFISH.py -ip 192.168.0.120 -u root -p calvin -o 2 -
d 1
```

this example will detach CD ISO image.

Optional arguments:

```
  -h, --help        show this help message and exit
  -ip IP            iDRAC IP address
  -u U              iDRAC username
  -p P              iDRAC password
  -c C              Get current virtual media information, pass in a value
                    of "y"
  -o O              Pass in the type of action you want to perform. Pass in
                    "1" to Insert or "2" to Eject
```

```
  -d D              Pass in the device you want to insert or eject. Pass in
                    "1" for CD or "2" for RemovableDisk
  -i I              Insert (attach) virtual media, pass in the HTTP or HTTPS
                    URI path of the remote image. Note: If attaching
                    removable disk, only supported file type is .img
```

2. Check the current attach status for virtual media:

```
C:\>InsertEjectVirtualMediaREDFISH.py -ip 192.168.0.120 -u root -p calvin -c y
- Virtual Media URIs detected
/redfish/v1/Managers/iDRAC.Embedded.1/VirtualMedia/RemovableDisk
/redfish/v1/Managers/iDRAC.Embedded.1/VirtualMedia/CD


- Detailed information for URI
"/redfish/v1/Managers/iDRAC.Embedded.1/VirtualMedia/RemovableDisk"

ImageName: None
Description: iDRAC Virtual Media Services Settings
Image: None
ConnectedVia: NotConnected
Name: Virtual Removable Disk
WriteProtected: None
Inserted: False
Id: RemovableDisk
MediaTypes: [u'USBStick']

- Detailed information for URI
 "/redfish/v1/Managers/iDRAC.Embedded.1/VirtualMedia/CD"

ImageName: None
Description: iDRAC Virtual Media Services Settings
Image: None
ConnectedVia: NotConnected
Name: Virtual CD
WriteProtected: None
Inserted: False
Id: CD
MediaTypes: [u'CD', u'DVD']
```

3. Attach an ISO image that is stored on an HTTP share:

```
C:\>InsertEjectVirtualMediaREDFISH.py -ip 192.168.0.120 -u root -p calvin -o
    1 -d 1 -i http://192.168.0.130/esxi_5u1.iso

- WARNING, insert(attached) "CD" virtual media device
"http://192.168.0.130/updates_http/esxi_5u1.iso"

- PASS, POST command passed to successfully insert(attached) CD media
- PASS, GET command passed to verify CD media successfully
inserted(attached)
```

4. If required, check status to verify if the virtual CD has attached:

```
C:\>InsertEjectVirtualMediaREDFISH.py -ip 192.168.0.120 -u root -p calvin -c y

- Virtual Media URIs detected

/redfish/v1/Managers/iDRAC.Embedded.1/VirtualMedia/RemovableDisk
/redfish/v1/Managers/iDRAC.Embedded.1/VirtualMedia/CD

- Detailed information for URI
"/redfish/v1/Managers/iDRAC.Embedded.1/VirtualMedia/RemovableDisk"

ImageName: None
Description: iDRAC Virtual Media Services Settings
Image: None
ConnectedVia: NotConnected
Name: Virtual Removable Disk
WriteProtected: True
Inserted: False
Id: RemovableDisk
MediaTypes: [u'USBStick']

- Detailed information for URI
"/redfish/v1/Managers/iDRAC.Embedded.1/VirtualMedia/CD"
ImageName: esxi_5u1.iso
Description: iDRAC Virtual Media Services Settings
Image: http://192.168.0.130/esxi_5u1.iso
ConnectedVia: URI
Name: Virtual CD
WriteProtected: True
Inserted: True
Id: CD
MediaTypes: [u'CD', u'DVD']
```

5. After virtual media is attached, configure a one-time boot device to boot from the attached virtual media image. Because the Redfish standard doesn't support a method to configure a one-time boot from virtual media, a Dell EMC OEM extension API is required for this step.

   The script example below uses the **SetNextOneTimeBootVirtualMediaDeviceOemREDFISH.py** Python script to set one-time boot from virtual CD and reboot the server immediately:

```
C:\>SetNextOneTimeBootVirtualMediaDeviceOemREDFISH.py -ip 192.168.0.120 -u root
-p calvin -d 1 -r y
- WARNING, setting next onetime boot device to Virtual CD
- PASS, successfully set next onetime boot device to Virtual CD
- WARNING, Current server power state is: On
- PASS, Command passed to gracefully power OFF server, 204 status code returned
- PASS, GET command passed to verify server is in OFF state
- PASS, Command passed to power ON server, 204 status code returned
- WARNING, System will now reboot and onetime boot to Virtual CD after POST
```

2.2 Enhanced PowerEdge hardware inventory

Performing hardware inventory is a key server management function used by IT automation to discover the capabilities of a given server. iDRA9 3.30.30.30 adds new APIs that expand the existing hardware inventory APIs, enabling IT automation to determine which controllers are installed in which PCIe slots and to obtain hardware details for Non-Volatile Dual-Inline Memory Modules (NVDIMMs) installed in 14G servers.

The script example below illustrates obtaining server PCIe device slot information by using the Python script `GetPCIeDeviceInventoryREDFISH.py`. This script can get either PCIe device or function information.

Note—It is recommended to view script help text first before executing the script to see supported arguments and examples.

```
C:>GetPCIeDeviceInventoryREDFISH.py -ip 192.168.0.120 -u root -p calvin -d
yy

- WARNING, server PCIe Device URIs for iDRAC 100.65.205.134

/redfish/v1/Systems/System.Embedded.1/PCIeDevice/59-0
/redfish/v1/Systems/System.Embedded.1/PCIeDevice/0-0
/redfish/v1/Systems/System.Embedded.1/PCIeDevice/0-28
/redfish/v1/Systems/System.Embedded.1/PCIeDevice/0-23
/redfish/v1/Systems/System.Embedded.1/PCIeDevice/94-0
/redfish/v1/Systems/System.Embedded.1/PCIeDevice/0-17
/redfish/v1/Systems/System.Embedded.1/PCIeDevice/3-0
/redfish/v1/Systems/System.Embedded.1/PCIeDevice/0-31
/redfish/v1/Systems/System.Embedded.1/PCIeDevice/25-0
/redfish/v1/Systems/System.Embedded.1/PCIeDevice/177-0
/redfish/v1/Systems/System.Embedded.1/PCIeDevice/24-0

- Detailed information for URI
"/redfish/v1/Systems/System.Embedded.1/PCIeDevice/59-0"

@odata.type: #PCIeDevice.v1_2_0.PCIeDevice
SKU: None
Assembly: {u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1/Assembly'}
Description: BOSS-S1 Adapter
Links: {u'PCIeFunctions': [{u'@odata.id':
u'/redfish/v1/Systems/System.Embedded.1/PCIeFunction/59-0-0'}], u'Chassis':
[{u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1'}],
u'PCIeFunctions@odata.count': 1, u'Chassis@odata.count': 1}
AssetTag: None
SerialNumber: None
@odata.id: /redfish/v1/Systems/System.Embedded.1/PCIeDevice/59-0
@odata.context: /redfish/v1/$metadata#PCIeDevice.PCIeDevice
Status: {u'HealthRollup': u'OK', u'State': u'Enabled', u'Health': u'OK'}
PartNumber: None
DeviceType: SingleFunction
Name: BOSS-S1 Adapter
```

```
Model: None
Manufacturer: Marvell Technology Group Ltd.
Id: 59-0
FirmwareVersion: 2.5.13.3016
@odata.etag: 1550851147

- Detailed information for URI
"/redfish/v1/Systems/System.Embedded.1/PCIeDevice/0-0"

@odata.type: #PCIeDevice.v1_2_0.PCIeDevice
SKU: None
Assembly: {u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1/Assembly'}
Description: Sky Lake-E DMI3 Registers
Links: {u'PCIeFunctions': [{u'@odata.id':
u'/redfish/v1/Systems/System.Embedded.1/PCIeFunction/0-0-0'}], u'Chassis':
[{u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1'}],
u'PCIeFunctions@odata.count': 1, u'Chassis@odata.count': 1}
AssetTag: None
SerialNumber: None
@odata.id: /redfish/v1/Systems/System.Embedded.1/PCIeDevice/0-0
@odata.context: /redfish/v1/$metadata#PCIeDevice.PCIeDevice
Status: {u'HealthRollup': u'OK', u'State': u'Enabled', u'Health': u'OK'}
PartNumber: None
DeviceType: SingleFunction
Name: Sky Lake-E DMI3 Registers
Model: None
Manufacturer: Intel Corporation
Id: 0-0
FirmwareVersion:
@odata.etag: 1550851148


- Detailed information for URI
"/redfish/v1/Systems/System.Embedded.1/PCIeDevice/0-28"

@odata.type: #PCIeDevice.v1_2_0.PCIeDevice
SKU: None
Assembly: {u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1/Assembly'}
Description: C620 Series Chipset Family PCI Express Root Port #5
Links: {u'PCIeFunctions': [{u'@odata.id':
u'/redfish/v1/Systems/System.Embedded.1/PCIeFunction/0-28-4'},
{u'@odata.id': u'/redfish/v1/Systems/System.Embedded.1/PCIeFunction/0-28-
0'}], u'Chassis': [{u'@odata.id':
u'/redfish/v1/Chassis/System.Embedded.1'}], u'PCIeFunctions@odata.count': 2,
u'Chassis@odata.count': 1}
AssetTag: None
SerialNumber: None
@odata.id: /redfish/v1/Systems/System.Embedded.1/PCIeDevice/0-28
@odata.context: /redfish/v1/$metadata#PCIeDevice.PCIeDevice
Status: {u'HealthRollup': u'OK', u'State': u'Enabled', u'Health': u'OK'}
PartNumber: None
```

```
DeviceType: MultiFunction
Name: C620 Series Chipset Family PCI Express Root Port #5
Model: None
Manufacturer: Intel Corporation
Id: 0-28
FirmwareVersion:
@odata.etag: 1550851148


- Detailed information for URI
"/redfish/v1/Systems/System.Embedded.1/PCIeDevice/0-23"

@odata.type: #PCIeDevice.v1_2_0.PCIeDevice
SKU: None
Assembly: {u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1/Assembly'}
Description: C620 Series Chipset Family SATA Controller [AHCI mode]
Links: {u'PCIeFunctions': [{u'@odata.id':
u'/redfish/v1/Systems/System.Embedded.1/PCIeFunction/0-23-0'}], u'Chassis':
[{u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1'}],
u'PCIeFunctions@odata.count': 1, u'Chassis@odata.count': 1}
AssetTag: None
SerialNumber: None
@odata.id: /redfish/v1/Systems/System.Embedded.1/PCIeDevice/0-23
@odata.context: /redfish/v1/$metadata#PCIeDevice.PCIeDevice
Status: {u'HealthRollup': u'OK', u'State': u'Enabled', u'Health': u'OK'}
PartNumber: None
DeviceType: SingleFunction
Name: C620 Series Chipset Family SATA Controller [AHCI mode]
Model: None
Manufacturer: Intel Corporation
Id: 0-23
FirmwareVersion:
@odata.etag: 1550851148

- Detailed information for URI
"/redfish/v1/Systems/System.Embedded.1/PCIeDevice/94-0"

@odata.type: #PCIeDevice.v1_2_0.PCIeDevice
SKU: None
Assembly: {u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1/Assembly'}
Description: Express Flash NVMe PM1725 6.4TB AIC
Links: {u'PCIeFunctions': [{u'@odata.id':
u'/redfish/v1/Systems/System.Embedded.1/PCIeFunction/94-0-0'}], u'Chassis':
[{u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1'}],
u'PCIeFunctions@odata.count': 1, u'Chassis@odata.count': 1}
AssetTag: None
SerialNumber: None
@odata.id: /redfish/v1/Systems/System.Embedded.1/PCIeDevice/94-0
@odata.context: /redfish/v1/$metadata#PCIeDevice.PCIeDevice
Status: {u'HealthRollup': u'OK', u'State': u'Enabled', u'Health': u'OK'}
PartNumber: None
```

```
DeviceType: SingleFunction
Name: Express Flash NVMe PM1725 6.4TB AIC
Model: None
Manufacturer: Samsung Electronics Co Ltd
Id: 94-0
FirmwareVersion: KPYABD3Q
@odata.etag: 1550851148


- Detailed information for URI
"/redfish/v1/Systems/System.Embedded.1/PCIeDevice/0-17"

@odata.type: #PCIeDevice.v1_2_0.PCIeDevice
SKU: None
Assembly: {u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1/Assembly'}
Description: C620 Series Chipset Family SSATA Controller [AHCI mode]
Links: {u'PCIeFunctions': [{u'@odata.id':
u'/redfish/v1/Systems/System.Embedded.1/PCIeFunction/0-17-5'}], u'Chassis':
[{u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1'}],
u'PCIeFunctions@odata.count': 1, u'Chassis@odata.count': 1}
AssetTag: None
SerialNumber: None
@odata.id: /redfish/v1/Systems/System.Embedded.1/PCIeDevice/0-17
@odata.context: /redfish/v1/$metadata#PCIeDevice.PCIeDevice
Status: {u'HealthRollup': u'OK', u'State': u'Enabled', u'Health': u'OK'}
PartNumber: None
DeviceType: MultiFunction
Name: C620 Series Chipset Family SSATA Controller [AHCI mode]
Model: None
Manufacturer: Intel Corporation
Id: 0-17
FirmwareVersion:
@odata.etag: 1550851148

- Detailed information for URI
"/redfish/v1/Systems/System.Embedded.1/PCIeDevice/3-0"

@odata.type: #PCIeDevice.v1_2_0.PCIeDevice
SKU: None
Assembly: {u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1/Assembly'}
Description: Integrated Matrox G200eW3 Graphics Controller
Links: {u'PCIeFunctions': [{u'@odata.id':
u'/redfish/v1/Systems/System.Embedded.1/PCIeFunction/3-0-0'}], u'Chassis':
[{u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1'}],
u'PCIeFunctions@odata.count': 1, u'Chassis@odata.count': 1}
AssetTag: None
SerialNumber: None
@odata.id: /redfish/v1/Systems/System.Embedded.1/PCIeDevice/3-0
@odata.context: /redfish/v1/$metadata#PCIeDevice.PCIeDevice
Status: {u'HealthRollup': u'OK', u'State': u'Enabled', u'Health': u'OK'}
PartNumber: None
DeviceType: SingleFunction
```

```
Name: Integrated Matrox G200eW3 Graphics Controller
Model: None
Manufacturer: Matrox Electronics Systems Ltd.
Id: 3-0
FirmwareVersion:
@odata.etag: 1550851148


- Detailed information for URI
"/redfish/v1/Systems/System.Embedded.1/PCIeDevice/0-31"

@odata.type: #PCIeDevice.v1_2_0.PCIeDevice
SKU: None
Assembly: {u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1/Assembly'}
Description: C621 Series Chipset LPC/eSPI Controller
Links: {u'PCIeFunctions': [{u'@odata.id':
u'/redfish/v1/Systems/System.Embedded.1/PCIeFunction/0-31-0'},
{u'@odata.id': u'/redfish/v1/Systems/System.Embedded.1/PCIeFunction/0-31-
4'}], u'Chassis': [{u'@odata.id':
u'/redfish/v1/Chassis/System.Embedded.1'}], u'PCIeFunctions@odata.count': 2,
u'Chassis@odata.count': 1}
AssetTag: None
SerialNumber: None
@odata.id: /redfish/v1/Systems/System.Embedded.1/PCIeDevice/0-31
@odata.context: /redfish/v1/$metadata#PCIeDevice.PCIeDevice
Status: {u'HealthRollup': u'OK', u'State': u'Enabled', u'Health': u'OK'}
PartNumber: None
DeviceType: MultiFunction
Name: C621 Series Chipset LPC/eSPI Controller
Model: None
Manufacturer: Intel Corporation
Id: 0-31
FirmwareVersion:
@odata.etag: 1550851148
```

- Detailed information for URI
"/redfish/v1/Systems/System.Embedded.1/PCIeDevice/25-0"

```
@odata.type: #PCIeDevice.v1_2_0.PCIeDevice
SKU: None
Assembly: {u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1/Assembly'}
Description: BCM57800 1-Gigabit Ethernet
Links: {u'PCIeFunctions': [{u'@odata.id':
u'/redfish/v1/Systems/System.Embedded.1/PCIeFunction/25-0-2'},
{u'@odata.id': u'/redfish/v1/Systems/System.Embedded.1/PCIeFunction/25-0-
1'}, {u'@odata.id': u'/redfish/v1/Systems/System.Embedded.1/PCIeFunction/25-
0-3'}, {u'@odata.id':
u'/redfish/v1/Systems/System.Embedded.1/PCIeFunction/25-0-0'}], u'Chassis':
[{u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1'}],
u'PCIeFunctions@odata.count': 4, u'Chassis@odata.count': 1}
AssetTag: None
```

```
SerialNumber: CN779216C300OT
@odata.id: /redfish/v1/Systems/System.Embedded.1/PCIeDevice/25-0
@odata.context: /redfish/v1/$metadata#PCIeDevice.PCIeDevice
Status: {u'HealthRollup': u'OK', u'State': u'Enabled', u'Health': u'OK'}
PartNumber: 0G8RPD
DeviceType: MultiFunction
Name: BCM57800 1-Gigabit Ethernet
Model: None
Manufacturer: Broadcom Inc. and subsidiaries
Id: 25-0
FirmwareVersion: 08.07.00
@odata.etag: 1550851148


- Detailed information for URI
"/redfish/v1/Systems/System.Embedded.1/PCIeDevice/177-0"

@odata.type: #PCIeDevice.v1_2_0.PCIeDevice
SKU: None
Assembly: {u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1/Assembly'}
Description: Express Flash PM1725b 1.6TB SFF
Links: {u'PCIeFunctions': [{u'@odata.id':
u'/redfish/v1/Systems/System.Embedded.1/PCIeFunction/177-0-0'}], u'Chassis':
[{u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1'}],
u'PCIeFunctions@odata.count': 1, u'Chassis@odata.count': 1}
AssetTag: None
SerialNumber: None
@odata.id: /redfish/v1/Systems/System.Embedded.1/PCIeDevice/177-0
@odata.context: /redfish/v1/$metadata#PCIeDevice.PCIeDevice
Status: {u'HealthRollup': u'OK', u'State': u'Enabled', u'Health': u'OK'}
PartNumber: None
DeviceType: SingleFunction
Name: Express Flash PM1725b 1.6TB SFF
Model: None
Manufacturer: Samsung Electronics Co Ltd
Id: 177-0
FirmwareVersion: 1.0.0
@odata.etag: 1550851148

- Detailed information for URI
"/redfish/v1/Systems/System.Embedded.1/PCIeDevice/24-0"

@odata.type: #PCIeDevice.v1_2_0.PCIeDevice
SKU: None
Assembly: {u'@odata.id': u'/redfish/v1/Chassis/System.Embedded.1/Assembly'}
Description: PERC H330 Mini
Links: {u'PCIeFunctions': [{u'@odata.id':
u'/redfish/v1/Systems/System.Embedded.1/PCIeFunction/24-0-0'}], u'Chassis':
[{u'@odata.id': u'/redfish/v1/Chassis/Enclosure.Internal.0-
1:RAID.Integrated.1-1'}], u'PCIeFunctions@odata.count': 1,
u'Chassis@odata.count': 1}
AssetTag: None
```

```
SerialNumber: CN7792174K03GE
@odata.id: /redfish/v1/Systems/System.Embedded.1/PCIeDevice/24-0
@odata.context: /redfish/v1/$metadata#PCIeDevice.PCIeDevice
Status: {u'HealthRollup': u'OK', u'State': u'Enabled', u'Health': u'OK'}
PartNumber: 0GDJ3J
DeviceType: SingleFunction
Name: PERC H330 Mini
Model: None
Manufacturer: LSI Logic / Symbios Logic
Id: 24-0
FirmwareVersion: 25.5.4.0006
@odata.etag: 1550851148

- WARNING, detailed information also captured in "pcie_devices.txt" file
```

The script example below displays NVDIMM information by using the Python script `GetNvDimmInventoryREDFISH.py`.

Note—It is recommended to view script help text before executing the script to see supported arguments and examples.

```
C:\>GetNvDimmInventoryREDFISH.py -ip 192.168.0.120 -u root -p calvin

- WARNING, NVDIMM URI(s) detected for iDRAC IP "100.65.99.162"

/redfish/v1/Systems/System.Embedded.1/Memory/iDRAC.Embedded.1%23DIMMSLOTA7

- Detailed NVDIMM information for URI
"/redfish/v1/Systems/System.Embedded.1/Memory/iDRAC.Embedded.1%23DIMMSLOTA7"
-

ModuleProductID: 0x4e32
NonVolatileSizeMiB: 16384
SerialNumber: 19B6C7A2
Status: {u'State': u'Enabled', u'Health': u'OK'}
Regions: []
OperatingMemoryModes: [u'PMEM']
VolatileSizeMiB: 0
MaxTDPMilliWatts: []
PartNumber: 18ASF2G72XF12G6V21AB
DeviceLocator: DIMM A7
Description: DIMM A7
LogicalSizeMiB: 0
AllowedSpeedsMHz: [2400]
MemoryMedia@odata.count: 0
BankLabel: A
ManufactureDate: Mon Nov 13 06:00:00 2017 UTC
Model: DDR4 DIMM
CacheSizeMiB: 0
OperatingSpeedMhz: 2400
Regions@odata.count: 0
Manufacturer: Micron Technology
OperatingMemoryModes@odata.count: 1
Name: DIMM A7
```

```
AllowedSpeedsMHz@odata.count: 1
BusWidthBits: 72
MemoryType: NVDIMM_N
FirmwareRevision: 9324
BaseModuleType: None
DataWidthBits: 64
MemoryMedia: []
RankCount: 1
MaxTDPMilliWatts@odata.count: 0
MemorySubsystemControllerManufacturerID: 0x3480
ModuleManufacturerID: 0x2c80
ErrorCorrection: MultiBitECC
MemorySubsystemControllerProductID: 0x4131
MemoryDeviceType: DDR4
Id: iDRAC.Embedded.1#DIMMSLOTA7
CapacityMiB: 16384
- WARNING, output also captured in "NVDIMM_inventory.txt" file
```

2.3 Configure the boot order

The rise of cloud computing has driven the need for automated support of server provisioning because servers more routinely change operating workloads. A typical automated server provisioning function will reconfigure the server boot source multiple time while performing hardware or firmware inventory, firmware update, hardware configuration, and OS deployment. iDRAC RESTful APIs in 3.30.30.30 now enable the user to either enable or disable boot order devices or change the boot order with standards-based APIs.

The script example below displays current boot option states and the state of boot option disables using the Python script **EnableDisableBootOrderDevicesDMTF_REDFISH.py**.

Note—It is recommended to view script help text before executing the script to see supported arguments and examples.

1. Get the boot order devices and their current enabled state. Notice that there are two boot order devices and both are in the enabled state (highlighted):

```
C:\Python27>EnableDisableBootOrderDevicesDMTF_REDFISH.py -ip 192.168.0.120 -
u root -p calvin -B y
- Boot Option URIs for "Uefi" Boot Mode -
/redfish/v1/Systems/System.Embedded.1/BootOptions/Boot0000
/redfish/v1/Systems/System.Embedded.1/BootOptions/Boot0001

- Detailed information for URI
/redfish/v1/Systems/System.Embedded.1/BootOptions/Boot0000 -

DisplayName: PCIe SSD in Slot 2: Windows Boot Manager
Description: Current settings of the UEFI Boot option
BootOptionEnabled: True
UefiDevicePath: HD(2,GPT,C39111CF-3093-43CF-AFC8-
6FBA426ED5EB,0xE1800,0x32000)/\EFI\Microsoft\Boot\bootmgfw.efi
BootOptionReference: Boot0000
Id: Boot0000
Name: Uefi Boot Option
```

```
- Detailed information for URI
/redfish/v1/Systems/System.Embedded.1/BootOptions/Boot0001 -

DisplayName: PXE Device 1: Integrated NIC 1 Port 1 Partition 1
Description: Current settings of the UEFI Boot option
BootOptionEnabled: True
UefiDevicePath: VenHw(3A191845-5F86-4E78-8FCE-C4CFF59F9DAA)
BootOptionReference: Boot0001
Id: Boot0001
Name: Uefi Boot Option
```

2. Select the Windows Boot Manager entry boot order device for disablement. The URI of this device will be provided to disable it (Example: URI "/redfish/v1/Systems/System.Embedded.1/BootOptions/Boot0000")

```
C:\Python27>EnableDisableBootOrderDevicesDMTF_REDFISH.py -ip 192.168.0.120 -
u root -p calvin -c
/redfish/v1/Systems/System.Embedded.1/BootOptions/Boot0000 -e false

- PASS: PATCH command passed to change boot option URI
"/redfish/v1/Systems/System.Embedded.1/BootOptions/Boot0000" to "false"
- PASS, job ID JID_512974957095 successfully created
- WARNING: JobStatus not scheduled, Current status is: New
- PASS, JID_512974957095 job id successfully scheduled, rebooting the server
to apply boot option changes

- WARNING, Current server power state is: On
- PASS, Command passed to gracefully power OFF server, status code return is
204
- PASS, GET command passed to verify server is in OFF state
- PASS, Command passed to power ON server, status code return is 204
- WARNING, JobStatus not completed, current status is: "Task successfully
scheduled."
. . .
- WARNING, JobStatus not completed, current status is: "Job in progress."

- Final detailed job results -
 JobID = JID_512974957095
 Name = ConfigBIOS:BIOS.Setup.1-1
 Message = Job completed successfully.
 PercentComplete = 100

- PASS, boot option enabled successfully set to "False" for URI
"/redfish/v1/Systems/System.Embedded.1/BootOptions/Boot0000"
```

3. The script validates the enable or disable state of boot devices, but if required, the script can be executed again to validate that the device was disabled (highlighted):

```
C:\Python27>EnableDisableBootOrderDevicesDMTF_REDFISH.py -ip 192.168.0.120 -
u root -p calvin -s
/redfish/v1/Systems/System.Embedded.1/BootOptions/Boot0000

- Detailed information for URI
/redfish/v1/Systems/System.Embedded.1/BootOptions/Boot0000 -
DisplayName: PCIe SSD in Slot 2: Windows Boot Manager
Description: Current settings of the UEFI Boot option
BootOptionEnabled: False
UefiDevicePath: HD(2,GPT,C39111CF-3093-43CF-AFC8-
6FBA426ED5EB,0xE1800,0x32000)/\EFI\Microsoft\Boot\bootmgfw.efi
BootOptionReference: Boot0000
Id: Boot0000
Name: Uefi Boot Option
```

The script example below obtains the current boot order and changes the boot order by using the Python script `ChangeBiosBootOrderDMTF_REDFISH.py`.

Note—It is recommended to view script help text before executing the script to see supported arguments and examples.

1. Get the current BIOS boot mode and the boot order:

```
C:\Python27>ChangeBiosBootOrderDMTF_REDFISH.py -ip 192.168.0.120 -u root -p
calvin -g y
- Current boot order detected for BIOS boot mode "Uefi" -
SequenceNumber: 0, DisplayName: PCIe SSD in Slot 2: Windows Boot Manager,
Id: Boot0000
SequenceNumber: 1, DisplayName: PXE Device 1: Integrated NIC 1 Port 1
Partition 1, Id: Boot0001
```

2. Use the script to change the boot order. To change the boot order, pass in the “Id” entries of the boot devices for the PATCH command. In this example, we want the PXE device to be first. Therefore, pass in boot device ID as “Boot0001”, and then “Boot0000”:

```
C:\Python27>ChangeBiosBootOrderDMTF_REDFISH.py -ip 192.168.0.120 -u root -p
calvin -c Boot0001,Boot0000
- PASS: PATCH command passed to change Uefi boot order sequence
- PASS, job ID "JID_512983746602" successfully created to change Uefi boot
order sequence
- WARNING: JobStatus not scheduled, current status is: New
- PASS, JID_512983746602 job id successfully scheduled, rebooting the server
to apply config changes

- WARNING, Current server power state is: On
- PASS, Command passed to gracefully power OFF server, code return is 204
- PASS, GET command passed to verify server is in OFF state
- PASS, Command passed to power ON server, code return is 204
- WARNING, JobStatus not completed, current status is: "Task successfully
scheduled."
. . .
```

```
- WARNING, JobStatus not completed, current status is: "Job in progress."
. . .
- WARNING, JobStatus not completed, current status is: "Job in progress."

- Final detailed job results -

 JobID = JID_512983746602
 Name = ConfigBIOS:BIOS.Setup.1-1
 Message = Job completed successfully.
 PercentComplete = 100

- Current boot order detected for BIOS boot mode "Uefi" -

SequenceNumber: 0, DisplayName: PXE Device 1: Integrated NIC 1 Port 1
Partition 1, Id: Boot0001
SequenceNumber: 1, DisplayName: PCIe SSD in Slot 2: Windows Boot Manager,
Id: Boot0000
```

2.4 Schedule maintenance window

In modern data centers, most configuration changes and updates that necessitate server downtime take place during a scheduled maintenance window. To support the automation of Redfish operations, new APIs using DMTF @Redfish.SettingsApplyTime have been added that enable the scheduling of Redfish operations that change the server configuration or perform a firmware update.

The script example below obtains one BIOS attribute—the “EmbSata” attribute—and changes the BIOS attribute current setting by using the Python script `GetSetBiosAttributesREDFISH.py`:

Note—It is recommended to view the script help text before executing the script to see supported arguments and examples.

---

1. Get the current value for attribute `EmbSata`:

```
C:\Python27>GetSetBiosAttributesREDFISH.py -ip 192.168.0.120 -u root -p
calvin -A EmbSata

- Current value for attribute "EmbSata" is "AhciMode"
```

2. Get the attribute registry information of this attribute. This will provide details about the type of the attribute, the attribute’s possible values, dependencies, and other details:

```
C:\Python27>GetSetBiosAttributesREDFISH.py -ip 192.168.0.120 -u root -p
calvin -s EmbSata
- WARNING, searching BIOS registry for attribute "EmbSata"
- Attribute Registry information for attribute "EmbSata" -
CurrentValue: None
DisplayName: Embedded SATA
MenuPath: ./SataSettingsRef
AttributeName: EmbSata
WarningText: None
```

```
Value: [{u'ValueDisplayName': u'AHCI Mode', u'ValueName': u'AhciMode'},
{u'ValueDisplayName': u'RAID Mode', u'ValueName': u'RaidMode'},
{u'ValueDisplayName': u'Off', u'ValueName': u'Off'}]
ReadOnly: False
WriteOnly: False
HelpText: Allows the Embedded SATA to be set to Off, AHCI, or RAID Mode.
Hidden: False
Type: Enumeration
Immutable: False
DisplayOrder: 5900
```

3. Set “EmbSata” to **RaidMode,** which is listed as one of the possible values. New for iDRAC9 3.30.30.30 firmware, you can use a standard Redfish API to schedule an “apply time” for the new attribute value within a defined maintenance window. In the below example, a BIOS configuration job is created to be executed on February 27th at 11PM; this job will reboot the server and apply the change. The API includes a duration time of 20 minutes; if the job does not complete within 20 minutes because of server issues such as failure to complete POST or failure of the OS to honor the server reboot request, the job will be marked as failed:

```
C:\Python27>GetSetBiosAttributesREDFISH.py -ip 192.168.0.120 -u root -p
calvin -an EmbSata -av RaidMode -r s -mt n -st 2019-02-27T23:00:00-06:00 -dt
1200
- WARNING, script will be setting BIOS attributes -
Attribute Name: EmbSata, setting new value to: RaidMode
- PASS: PATCH command passed to set BIOS attribute pending values and create
maintenance window config job, status code 202 returned
--- PASS, Detailed Job Status Results ---
JobState: New
Description: Job Instance
CompletionTime: None
PercentComplete: 0
StartTime: 2019-02-27T23:00:00
MessageId: JCP000
Message: New
EndTime: 2019-02-27T23:20:00
Id: JID_513083287277
JobType: BIOSConfiguration
Name: ConfigBIOS:BIOS.Setup.1-1

- PASS JID_513083287277 maintenance window config JID successfully created.
```

The job will go to scheduled state after the start time has elapsed and will automatically reboot the server to apply the configuration change.

2.5 Privilege registry

Service Processors, such as iDRAC9, typically provide a means to define the privileges afforded to an authenticated user. These privileges define which operations a given user can perform or restricts the set of objects such as storage controllers or NICs on which operations can be performed.

The newly added Redfish Privilege Registry enables IT administrators to map authenticated Redfish API users to available server resources and operations, enabling administrators to control access to specific devices and restrict operations on those devices as needed.

The following Python scripts illustrate using the Privilege Registry to determine user access and control permissions for access to specific devices or specific operations on those devices.

The script example below obtains the privileges for each entity (schema) that the iDRAC supports using the Python script `GetSchemaPrivilegesREDFISH.py`. For each schema, the script will return the type of commands supported along with which privileges are required to execute the commands. Note: the example below will be an edited version of output from the script because of the large amount of data that is returned.

Note—It is recommended to view script help text first before executing the script to view the supported arguments and examples.

```
C:\Python27>GetSchemaPrivilegesREDFISH.py -ip 192.168.0.120 -u root -p
calvin
- Privileges for Entity(Schema) "Redundancy" -
HEAD: [{u'Privilege': [u'Login'], u'Privilege@odata.count': 1}]
GET: [{u'Privilege': [u'Login'], u'Privilege@odata.count': 1}]
GET@odata.count: 1
PATCH: [{u'Privilege': [u'Login', u'ConfigureComponents'],
u'Privilege@odata.count': 2}]
POST@odata.count: 1
HEAD@odata.count: 1
PATCH@odata.count: 1
POST: [{u'Privilege': [u'Login', u'ConfigureComponents'],
u'Privilege@odata.count': 2}]

- Privileges for Entity(Schema) "DellPersistentStorageService" -

HEAD: [{u'Privilege': [u'Login'], u'Privilege@odata.count': 1}]
GET: [{u'Privilege': [u'Login'], u'Privilege@odata.count': 1}]
GET@odata.count: 1
POST@odata.count: 1
POST: [{u'Privilege': [u'AccessVirtualMedia', u'Login'],
u'Privilege@odata.count': 2}]
HEAD@odata.count: 1

- Privileges for Entity(Schema) "ServiceRoot" -

HEAD: [{u'Privilege': [], u'Privilege@odata.count': 0}]
GET@odata.count: 1
HEAD@odata.count: 1
```

```
GET: [{u'Privilege': [], u'Privilege@odata.count': 0}]

- Privileges for Entity(Schema) "DellPowerSupplyView" -

HEAD: [{u'Privilege': [u'Login'], u'Privilege@odata.count': 1}]
GET@odata.count: 1
HEAD@odata.count: 1
GET: [{u'Privilege': [u'Login'], u'Privilege@odata.count': 1}]

- Privileges for Entity(Schema) "DellEnclosureTemperatureSensor" -

HEAD: [{u'Privilege': [u'Login'], u'Privilege@odata.count': 1}]
GET@odata.count: 1
HEAD@odata.count: 1
GET: [{u'Privilege': [u'Login'], u'Privilege@odata.count': 1}]

- Privileges for Entity(Schema) "Role" -

HEAD: [{u'Privilege': [u'Login'], u'Privilege@odata.count': 1}]
GET: [{u'Privilege': [u'Login'], u'Privilege@odata.count': 1}]
GET@odata.count: 1
PATCH: [{u'Privilege': [u'ConfigureManager', u'Login'],
u'Privilege@odata.count': 2}]
POST@odata.count: 1
HEAD@odata.count: 1
PATCH@odata.count: 1
POST: [{u'Privilege': [u'ConfigureManager', u'Login'],
u'Privilege@odata.count': 2}]

- Privileges for Entity(Schema) "TaskCollection" -

HEAD: [{u'Privilege': [u'Login'], u'Privilege@odata.count': 1}]
GET@odata.count: 1
HEAD@odata.count: 1
GET: [{u'Privilege': [u'Login'], u'Privilege@odata.count': 1}]

- Privileges for Entity(Schema) "DellVirtualDisk" -

HEAD: [{u'Privilege': [u'Login'], u'Privilege@odata.count': 1}]
GET@odata.count: 1
HEAD@odata.count: 1
GET: [{u'Privilege': [u'Login'], u'Privilege@odata.count': 1}]
```

2.6 Host interface

Redfish 2017 added APIs to obtain access to a server-local interface for the iDRAC. These APIs are intended for use by future OSs that will provide server-local applications access to the service processor's Redfish implementation with a device driver or other methods.

GET on `URI`
`redfish/v1/Managers/iDRAC.Embedded.1/HostInterfaces/iDRAC.Embedded.1%23Host.`

`1` will return host interface information for the iDRAC. This GET returns the settings for the OS-to-iDRAC pass-through interface.

Example of output returned for GET:

```
@odata.context "/redfish/v1/$metadata#HostInterface.HostInterface"
@odata.id
   "/redfish/v1/Managers/iDRAC.Embedded.1/HostInterfaces/iDRAC.Embedded.1%23
Host.1"
@odata.type    "#HostInterface.v1_1_1.HostInterface"
Description    "Management for Host Interface"
ExternallyAccessible false
HostInterfaceType    "NetworkHostInterface"
Id "iDRAC.Embedded.1#Host.1"
InterfaceEnabled     true
Name     "Managed Host Interface 1"
```

2.7 Redfish SimpleUpdate API enhancements

iDRAC9 3.30.30.30 also enhances support for the Redfish standard API SimpleUpdate, adding the **TransferProtocol** option. Using this new method, a single device firmware update can be performed by providing the URL for an HTTP based Dell Update Package (DUP) file. When the SimpleUpdate with TransferProtocol API is executed via a POST, the DUP is downloaded to the iDRAC, an update job created, and the job scheduled to run.

The script example below uses the Python script `DeviceFirmwareSimpleUpdateTransferProtocolREDFISH.py` to perform a BIOS update by using a DUP package copied to an HTTP share.

Note—It is recommended to view script help text first before executing the script to see the supported arguments and examples.

1. Determine which protocols are supported with this version of iDRAC. For iDRAC9 3.30.30.30 firmware, only HTTP is supported by the TransferProtcol; additional protocols will be supported in future iDRAC9 releases:

```
C:\Python27>DeviceFirmwareSimpleUpdateTransferProtocolREDFISH.py -ip
192.168.0.120 -u root -p calvin -s y

- Supported protocols for TransferProtocol parameter (-t argument) - HTTP
```

2. Check the current version of BIOS installed on the server—BIOS version 1.6.13 is reported:

```
C:\Python27>DeviceFirmwareSimpleUpdateTransferProtocolREDFISH.py -ip
192.168.0.120 -u root -p calvin -g y

- WARNING, current devices detected with firmware version and updateable
status
```

```
Device Name: PCIe SSD in Slot 2 in Bay 1, Firmware Version: 1.0.0,
Updatable: True
Device Name: BOSS-S1, Firmware Version: 2.5.13.3016, Updatable: True
Device Name: PERC H330 Mini, Firmware Version: 25.5.5.0005, Updatable: True
Device Name: OS Collector, Firmware Version: 0, Updatable: True
Device Name: PCIe SSD in Slot 2, Firmware Version: KPYABD3Q, Updatable: True
Device Name: BP14G+EXP 0:1, Firmware Version: 2.40, Updatable: True
Device Name: iDRAC Service Module Installer, Firmware Version: 0, Updatable:
True
Device Name: Power Supply.Slot.1, Firmware Version: 00.23.32, Updatable:
True
Device Name: Disk 0 on AHCI Controller in slot 1, Firmware Version: DL43,
Updatable: True
Device Name: BIOS, Firmware Version: 1.6.13, Updatable: True
Device Name: Dell OS Driver Pack, 18.12.00, A00, Firmware Version: 18.12.00,
Updatable: True
Device Name: Disk 1 in Backplane 1 of Integrated RAID Controller 1, Firmware
Version: K774, Updatable: True
Device Name: Integrated Dell Remote Access Controller, Firmware Version:
3.30.30.30, Updatable: True
Device Name: Dell 64 Bit uEFI Diagnostics, version 4301, 4301A25, 4301.26,
Firmware Version: 4301A25, Updatable: True
Device Name: Disk 0 in Backplane 1 of Integrated RAID Controller 1, Firmware
Version: FSF9, Updatable: True
Device Name: QLogic 577xx/578xx 10 Gb Ethernet BCM57800 - 18:66:DA:8E:28:24,
Firmware Version: 08.07.00, Updatable: True
Device Name: System CPLD, Firmware Version: 1.0.2, Updatable: True
Device Name: Lifecycle Controller, Firmware Version: 3.30.30.30, Updatable:
False
```

3. With the BIOS DUP already copied to the HTTP share, pass in the URI path of the HTTP share to perform the update to BIOS version 1.7.24:

```
C:\Python27>DeviceFirmwareSimpleUpdateTransferProtocolREDFISH.py -ip
192.168.0.120 -u root -p calvin -t HTTP --uri
http://192.168.0.120/updates_http/BIOS_WN64_1.7.24.EXE -r y
- PASS, JID_517461677603 firmware update job ID successfully created for
update image "BIOS_WN64_1.7.24.EXE"
- Message: New, current job execution time is: 0:00:00
- Message: New, current job execution time is: 0:00:02
- Message: Downloading the BIOS_0XGXR_WN64_1.7.24.EXE update package.,
current job execution time is: 0:00:04
- Message: Downloading the BIOS_0XGXR_WN64_1.7.24.EXE update package.,
current job execution time is: 0:00:06
- Message: Downloading the BIOS_0XGXR_WN64_1.7.24.EXE update package.,
current job execution time is: 0:00:08
- Message: Package successfully downloaded., current job execution time is:
0:00:10
- PASS, job ID successfully marked as scheduled, powering on or rebooting
the server to apply the update
```

```
- WARNING, Current server power state is: On
- PASS, Command passed to gracefully power OFF server, code return is 204
- PASS, GET command passed to verify server is in OFF state
- PASS, Command passed to power ON server, code return is 204
- WARNING, JobStatus not completed, current status is: "Task successfully
scheduled.", job execution time is "0:00:12"
. . .
- WARNING, JobStatus not completed, current status is: "Task successfully
scheduled.", job execution time is "0:00:37"
. . .
- WARNING, JobStatus not completed, current status is: "The specified job is
in progress.", job execution time is "0:06:39"

- PASS, job ID JID_517461677603 successfully marked completed

- Final detailed job results -

@odata.type: #DellJob.v1_0_1.DellJob
JobState: Completed
Description: Job Instance
TargetSettingsURI: None
MessageArgs: []
CompletionTime: 2019-03-04T18:43:22
PercentComplete: 100
StartTime: TIME_NOW
MessageId: PR19
Message: The specified job has completed successfully.
EndTime: TIME_NA
Id: JID_517461677603
JobType: FirmwareUpdate
Name: Firmware Update

- JOB ID JID_517461677603 completed in 0:06:51
```

4. Verify the newly installed version of the BIOS:

```
C:\Python27>DeviceFirmwareSimpleUpdateTransferProtocolREDFISH.py -ip
192.168.0.120 -u root -p calvin -g y

- WARNING, current devices detected with firmware version and updateable
status

Device Name: PCIe SSD in Slot 2 in Bay 1, Firmware Version: 1.0.0,
Updatable: True
Device Name: BOSS-S1, Firmware Version: 2.5.13.3016, Updatable: True
Device Name: PERC H330 Mini, Firmware Version: 25.5.5.0005, Updatable: True
Device Name: OS Collector, Firmware Version: 0, Updatable: True
Device Name: PCIe SSD in Slot 2, Firmware Version: KPYABD3Q, Updatable: True
Device Name: BP14G+EXP 0:1, Firmware Version: 2.40, Updatable: True
```

```
Device Name: iDRAC Service Module Installer, Firmware Version: 0, Updatable:
True
Device Name: Power Supply.Slot.1, Firmware Version: 00.23.32, Updatable:
True
Device Name: Disk 0 on AHCI Controller in slot 1, Firmware Version: DL43,
Updatable: True
Device Name: BIOS, Firmware Version: 1.7.24, Updatable: True
Device Name: Dell OS Driver Pack, 18.12.00, A00, Firmware Version: 18.12.00,
Updatable: True
Device Name: Disk 1 in Backplane 1 of Integrated RAID Controller 1, Firmware
Version: K774, Updatable: True
Device Name: Integrated Dell Remote Access Controller, Firmware Version:
3.30.30.30, Updatable: True
Device Name: Dell 64 Bit uEFI Diagnostics, version 4301, 4301A25, 4301.26,
Firmware Version: 4301A25, Updatable: True
Device Name: Disk 0 in Backplane 1 of Integrated RAID Controller 1, Firmware
Version: FSF9, Updatable: True
Device Name: QLogic 577xx/578xx 10 Gb Ethernet BCM57800 - 18:66:DA:8E:28:24,
Firmware Version: 08.07.00, Updatable: True
Device Name: System CPLD, Firmware Version: 1.0.2, Updatable: True
Device Name: Lifecycle Controller, Firmware Version: 3.30.30.30, Updatable:
False
```

2.8 GET query parameters

Redfish 2017 R3 added support for query parameters as a part of a GET request. With the addition of the parameter, GET requests are transformed into a powerful query tool. Supported queries include:

- `$expand`—Returns data from subordinate resources, especially Collections, enabling highly-efficient data retrieval
- `$select`—Returns data with only the specified properties
- `$filter`—Returns data with only members of a collection that match an expression

Note—For the iDRAC9 3.30.30.30 release, $expand and $filter are not supported on Dell OEM extension URIs. That support is planned for a future release.

As an example, in the following Python code, a GET on PowerEdge server firmware inventory is fetched from the iDRAC, limiting the returned data to the top level with `$expand`:

```
req =
requests.get('https://%s/redfish/v1/UpdateService/FirmwareInventory?$expand=
*($levels=1)' % (idrac_ip), auth=(idrac_username, idrac_password),
verify=False)
```

In the following, $select is used to limit the data returned to those devices with the “MediaType” property when querying a set of drives:

```
req =
requests.get('https://%s/redfish/v1/Systems/System.Embedded.1/Storage/Drives
```

```
/Disk.Bay.1:Enclosure.Internal.0-1:RAID.Integrated.1-1?$select=MediaType' %
(idrac_ip), auth=(idrac_username, idrac_password), verify=False)
```

In the following, `$filter` is used to select a subset of entries in a resource collection which satisfies the filter criteria—it does not add any new data:

```
req =
requests.get('https://%s/redfish/v1/Systems/System.Embedded.1/Storage?$filte
r=@odata.id eq
'/redfish/v1/Systems/System.Embedded.1/Storage/AHCI.Embedded.1-1' %
(idrac_ip), auth=(idrac_username, idrac_password), verify=False)
```

Returns:

```
@odata.context "/redfish/v1/$metadata#StorageCollection.StorageCollection"
@odata.id
   "/redfish/v1/Systems/System.Embedded.1/Storage?$filter=@odata.id%20eq%20%
27/redfish/v1/Systems/System.Embedded.1/Storage/AHCI.Embedded.1-1%27"

@odata.type    "#StorageCollection.StorageCollection"
Description    "Collection Of Storage entities"
Members
0
@odata.id
   "/redfish/v1/Systems/System.Embedded.1/Storage/AHCI.Embedded.1-1"
Members@odata.count
1
Name     "Storage Collection"
```

Note—Applying filtering on URI "/redfish/v1/Systems/System.Embedded.1/Storage/AHCI.Embedded.1-1" will return only this subset from URI "redfish/v1/Systems/System.Embedded.1".

For more information about query parameters, refer to the DMTF Redfish API specification available at https://www.dmtf.org/.

3 Map iDRAC RESTful API to WS-Man

The Web Services Management (WS-Man) standard was established in 2010 by the DMTF to provide a structured method for describing server management data and operations. Based on the SOAP methodology, WS-Man was implemented by Dell EMC and other server vendors in their service processors starting from 2011. iDRAC supports a very robust set of WS-Man APIs that provide full lifecycle server management functions. These APIs have been supported from the 11G of PowerEdge through the current 14G PowerEdge servers.

Since the inception of WS-Man, however, RESTful APIs have risen to replace SOAP APIs as the preferred method for implementing API operations. As Redfish and the iDRAC RESTful API have matured, market demand has shifted from WS-Man to Redfish, leading to the deprecation of WS-Man in iDRAC9 3.30.30.30.

Given that developers have created significant software based on WS-Man, Dell EMC is building a bridge for WS-Man developers, providing iDRAC RESTful APIs that map to the WS-Man APIs. This effort will be delivered in several phases, starting with phase I in iDRAC9 3.30.30.30. Phase I will provide a broad range of iDRAC RESTful APIs for specific WS-Man classes. For more information, see **Appendix A Map iDRAC RESTful API to WS-Man Phase I** later in this white paper.

3.1 DellJobService

The DellJobService resource provides actions to support Job management functionality.

URI: `/redfish/v1/Dell/Managers/iDRAC.Embedded.1/DellJobService`

3.1.1 DeleteJobQueue

The DeleteJobQueue method is used for deleting jobs from the JobQueue or the job store. To clear all the jobs, use the keyword JID_CLEARALL for the JobID. Successful execution of this method with the JID_CLEARALL parameter value also clears all the pending attribute values. When the number of jobs in the JobQueue reaches the maximum limit, jobs in the "Completed" state are deleted automatically.

Note—On the 11G PowerEdge servers, jobs in the "Failed" state are not deleted automatically and must be removed manually one at a time, or all together by using the keyword `JID_CLEARALL`.

DeleteJobQueue Method Parameters

Qualifiers	Name	Type	Description
IN, Required	JobID	Edm.String	The JobID parameter specifies the job to be deleted. The value "JID_CLEARALL" for the JobID clears all the jobs.
OUT	Message	Edm.String	Error or information message corresponding to the MessageID is returned, in English.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.1.2 SetupJobQueue

This method is used for creating a job queue that shall contain one or more DellJobs with a specified order of execution within the queue.

SetupJobQueue Method Parameters

Qualifiers	Name	Type	Description
IN, Required	JobArray	Edm.String	The JobArray parameter will contain the array of JobIDs which represent the set of jobs to add to the job queue. This is an ordered array that represents the sequence in which the jobs are run.
IN, Required	StartTimeInterval	Edm.String	Start time for the job execution in the format `yyyymmddhhmmss`. The string "TIME_NOW" indicates immediate start.
IN	UntilTime	Edm.String	End time for the job execution in the format `yyyymmddhhmmss`. If this parameter is not NULL then the `StartTimeInterval` parameter must be specified.
OUT	Message	Edm.String	Error or information message corresponding to the MessageID is returned, in English.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.2 DellLCService

The DellLCService resource provides actions to support the Lifecycle Controller functionality.

URI: `/redfish/v1/Dell/Managers/iDRAC.Embedded.1/DellLCService`

3.2.1 BackupImage

The BackupImage method is used for backing up the firmware and configurations for Lifecycle Controller.

BackupImage Method Parameters

Qualifiers	Name	Type	Description
IN	CheckBackupDestinationAvailability	Edm.String	Check backup destination availability for CIFS, NFS, or HTTP, HTTPS share. Default is 0–Check.
IN	IPAddress	Edm.String	The IP address for the network share for the backup image. This is a required parameter when ShareType is specified as NFS or CIFS.
IN	IgnoreCertWarning	Edm.String	Specifies if certificate warning should be ignored when HTTPS is specified. Default is 2 (On).
IN	ImageName	Edm.String	Name of the image file.
IN	JobName	Edm.String	A name for the job. Default is 'Backup:Image'.

IN	Passphrase	Edm.String	The passphrase for the backup image. This parameter is required if ShareType is NFS or CIFS.
IN	Password	Edm.String	The password to access the network share.
IN	ProxyPasswd	Edm.String	The password for the proxy server.
IN	ProxyPort	Edm.String	Port for the proxy server. Default is set to 80.
IN	ProxyServer	Edm.String	The IP address of the proxy server.
IN	ProxySupport	Edm.String	Specifies if proxy should be used. Default is 1 (Off).
IN	ProxyType	Edm.String	The proxy type of the proxy server. Default is 0 (HTTP).
IN	ProxyUname	Edm.String	The user name for the proxy server.
IN	ScheduledStartTime	Edm.String	Schedules the job at the time specified. The format is yyyymmddhhmmss. Default value is TIME_NOW, which will start the job immediately.
IN	ShareName	Edm.String	Name of the CIFS share or full path to the NFS share. Optional for HTTP/HTTPS share, which may be treated as the path of the directory containing the file.
IN	ShareType	Edm.String	Type of the network share. Default is NFS, when this parameter is not passed.
IN	UntilTime	Edm.String	Scheduled end time for job execution in datetime format: `yyyymmddhhmmss`. If not specified, there is no end time.
IN	UserName	Edm.String	The user name to access the network share.
IN	Workgroup	Edm.String	Workgroup for the CIFS share - optional.
OUT	Job	Edm.String	Reference to the job spawned if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message corresponding to the message ID is returned, in English.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.2.2 ClearProvisioningServer

The ClearProvisioningServer method is used for clearing the provisioning server values.

ClearProvisioningServer Method Parameters

Qualifiers	Name	Type	Description
OUT	Message	Edm.String	Error or information message corresponding to the message ID is returned, in English.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.2.3 ExportFactoryConfiguration

The ExportFactoryConfiguration method is used for exporting the factory configuration from the Lifecycle Controller to a remote shared folder.

ExportFactoryConfiguration Method Parameters

Qualifiers	Name	Type	Description
IN	FileName	Edm.String	The target output file name. A file name is not required if the share type is Local.
IN	IPAddress	Edm.String	IP address of the network share, but not required if the share type is Local.
IN	IgnoreCertWarning	Edm.String	Specifies if certificate warning must be ignored when HTTPS is specified. If IgnoreCertWarning is On, warnings are ignored. Default is 2 (On).
IN	Password	Edm.String	Password of the account to access the share.
IN	ProxyPasswd	dm.String	The password for the proxy server.
IN	ProxyPort	Edm.String	Port for the proxy server. Default is set to 80.
IN	roxyServer	Edm.String	The IP address of the proxy server.
IN	ProxySupport	Edm.String	Specifies if proxy should be used. Default is 1 (Off).
IN	ProxyType	Edm.String	The proxy type of the proxy server. Default is 0 (HTTP).
IN	ProxyUname	Edm.String	The user name for the proxy server.
IN	ShareName	Edm.String	Name of the CIFS share or full path to the NFS share. Optional for HTTP/HTTPS share, which may be treated as the path of the directory containing the file.
IN, Required	ShareType	Edm.String	Type of the network share.
IN	UserName	Edm.String	User name of the account to access the share.
IN	Workgroup	Edm.String	Optional. Workgroup for the CIFS share.
OUT	Job	Edm.String	Reference to the job spawned if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message corresponding to the MessageID is returned, in English.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.2.4 ExportHWInventory

The ExportHWInventory method is used for exporting the hardware inventory from the Lifecycle Controller to a remote shared folder.

ExportHWInventory Method Parameters

Qualifiers	Name	Type	Description
IN	FileName	Edm.String	The arget output file name. A file name is not required if the share type is Local.
IN	IPAddress	Edm.String	IP address of the network share. An IP address is not required if the share type is Local.
IN	IgnoreCertWarning	Edm.String	Specifies if certificate warning must be ignored when HTTPS is specified. If IgnoreCertWarning is On, warnings are ignored. Default is 2 (On).
IN	Password	Edm.String	Password of the account to access the share.
IN	ProxyPasswd	Edm.String	The password for the proxy server.
IN	ProxyPort	Edm.String	Port for the proxy server. Default is set to 80.
IN	ProxyServer	Edm.String	The IP address of the proxy server.
IN	ProxySupport	Edm.String	Specifies if proxy should be used. Default is 1 (Off).
IN	ProxyType	Edm.String	The proxy type of the proxy server. Default is 0 (HTTP).
IN	ProxyUname	Edm.String	The user name for the proxy server.
IN	ShareName	Edm.String	Name of the CIFS share or full path to the NFS share. Optional for HTTP/HTTPS share (if supported), this may be treated as the path of the directory containing the file. ShareName is not required if share type is Local.
IN, Required	ShareType	Edm.String	Type of the network share.
IN	UserName	Edm.String	User name of the account to access the share.
IN	Workgroup	Edm.String	Optional. Workgroup for the CIFS share.
IN	XMLSchema	Edm.String	Type of XML output format. Default is CIM-XML.
OUT	Job	Edm.String	Reference to the job spawned if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message corresponding to the MessageID is returned, in English.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.2.5 ExportLCLog

The ExportLCLog method is used for exporting the log from the Lifecycle Controller to a remote shared folder.

ExportLCLog Method Parameters

Qualifiers	Name	Type	Description
IN	FileName	Edm.String	The target output file name. A file name is not required if the share type is Local.
IN	IPAddress	Edm.String	IP address of the network share. An IP address is not required if share type is Local.
IN	IgnoreCertWarning	Edm.String	Specifies if certificate warning must be ignored when HTTPS is specified. If IgnoreCertWarning is On, warnings are ignored. Default is 2 (On).
IN	Password	Edm.String	Password of the account to access the share.
IN	ProxyPasswd	Edm.String	The password for the proxy server.
IN	ProxyPort	Edm.String	Port for the proxy server. Default is set to 80.
IN	ProxyServer	Edm.String	The IP address of the proxy server.
IN	ProxySupport	Edm.String	Specifies if proxy must be used. Default is 1 (Off).
IN	ProxyType	Edm.String	The proxy type of the proxy server. Default is 0 (HTTP).
IN	ProxyUname	Edm.String	The user name for the proxy server.
IN	ShareName	Edm.String	Name of the CIFS share or full path to the NFS share. Optional for HTTP/HTTPS shares (if supported), this may be treated as the path of the directory containing the file.
IN, Required	ShareType	Edm.String	Type of the network share.
IN	UserName	Edm.String	UserName of the account to access the share.
IN	Workgroup	Edm.String	Optional. Workgroup for the CIFS share.
OUT	Job	Edm.String	Reference to the job spawned if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message corresponding to the message ID is returned, in English.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.2.6 ExportTechSupportReport

This method is used to collect the TSR that includes hardware, OS, and application data. The data is compressed into a compressed file and saved on the remote share (CIFS, NFS, HTTP, or HTTPS).

ExportTechSupportReport Method Parameters

Qualifiers	Name	Type	Description
IN	DataSelectorArrayIn	Array of Edm.string	Array of integer values to select TSR components.
IN	FileName	Uint16[]	File name of the export tech support report.
IN, Required	IPAddress	Edm.String	IP address of the network share.
IN	Password	Edm.String	Password of the account to access the share.
IN, Required	ShareName	Edm.String	Name of the CIFS share or full folder path to the NFS share. Optional for HTTP/HTTPS share (if supported), which may be treated as the path of the directory containing the file.
IN, Required	ShareType	Edm.String	Type of the network share.
IN	UserName	Edm.String	User name of the account to access the share.
OUT	Job	Edm.String	Reference to the job spawned if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message corresponding to the MessageID is returned, in English.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.2.7 GetRSStatus

The GetRSStatus method is used for obtaining the Data Manager (Remote Services) status.

Note—The GetRemoteServicesAPIStatus method execution reports more granular and detailed status of the Remote Services API.

GetRSStatus Method Parameters

Qualifiers	Name	Type	Description
OUT	Message	Edm.String	Error or information message corresponding to the Message ID is returned, in English.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	status	Edm.String	The status for the Data Manager—Ready, Not Ready, and Reloading.

3.2.8 GetRemoteServicesAPIStatus

The GetRemoteServicesAPIStatus method is used for obtaining the overall remote services API status that includes host system status, the remote services status, and real-time status. The overall rolled-up status is reflected in the Status output parameter. Note: The LCStatus output parameter value includes the status reported by the DMStatus output parameter in the GetRSStatus method and the Lifecycle Controller status. Thus, GetRSStatus method invocation is redundant.

GetRemoteServicesAPIStatus Method Parameters

Qualifiers	Name	Type	Description
OUT	LCStatus	Edm.String	The Lifecycle Controller status that includes the Data Manager status.
OUT	Message	Edm.String	Error or information message corresponding to the MessageID is returned, in English.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	RTStatus	Edm.String	The RealTime Status of the host server.
OUT	ServerStatus	Edm.String	Status of the host server.
OUT	Status	Edm.String	The overall status of the Remote Services API.

3.2.8.1 Using GetRemoteServicesAPIStatus

As a best practice, before initiating Redfish workflows that change the server's or iDRAC's configuration or that perform firmware update, the GetRemoteServicesAPIStatus API should be used. This call verifies that the server and iDRAC are ready and in a good state before initiating a workflow. In addition, if the workflow performs multiple operation steps that require server reboot between steps, this API should be used to ensure server and iDRAC readiness before the next step proceeds.

The script example below uses the Python script **GetRemoteServicesAPIStatusREDFISH.py** to check the status of the iDRAC remote services.

Note—It is recommended to view script help text first before executing the script to see the supported arguments and examples.

```
C:\Python27>GetRemoteServicesAPIStatusREDFISH.py -ip 192.168.0.120 -u root -
p calvin
```

```
-PASS: POST command passed for GetRemoteServicesAPIStatus method, status
code 200 returned

RTStatus: Ready
Status: Ready
LCStatus: Ready
ServerStatus: OutOfPOST
```

3.2.9 LCWipe

The LCWipe method is used for deleting all configurations from Lifecycle Controller before the system is retired. Host must be manually rebooted for the changes to take effect.

LCWipe Method Parameters

Qualifiers	Name	Type	Description
OUT	Message	Edm.String	Error or information message corresponding to the MessageID is returned, in English.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.2.10 ReInitiateDHS

The ReInitiateDHS method is used for reinitiating the provisioning server discovery and handshake.

ReInitiateDHS Method Parameters

Qualifiers	Name	Type	Description
IN, Required	PerformAutoDiscovery	Edm.String	A value of "Off = 1" disables auto discovery. A value of "Now = 2" enables and initiates auto discovery immediately. A value of "NextBoot = 3" delays reconfiguration and auto discovery until next AC powercycle. **Note**—If NextBoot has a value of 3 then after successful execution, the Discovery Factory Defaults attribute is set to "On".
IN	ProvisioningServer	Edm.String	This property specifies the provisioning server addresses and ports used for auto discovery. If omitted, Lifecycle Controller gets the value from DHCP or DNS.
IN, Required	ResetToFactoryDefaults	Edm.Boolean	If ResetToFactoryDefaults is True, all configuration information is replaced with auto-discovery factory default properties. If False, an error is returned.
OUT	Message	Edm.String	Error or information message corresponding to the Message ID is returned, in English.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.2.11 RestoreImage

This RestoreImage method is used to restore firmware and configurations.

RestoreImage Method Parameters

Qualifiers	Name	Type	Description
IN	IPAddress	Edm.String	IP address of the NFS or CIFS share. The parameter is required if the ShareType parameter has value 0 (NFS), 2 (CIFS), or is not specified.
IN	IgnoreCertWarning	Edm.String	Specifies if certificate warning should be ignored when HTTPS is specified. If IgnoreCertWarning is On, warnings are ignored. Default is 2 (On).
IN	ImageName	Edm.String	Name of the image file.
IN	Passphrase	Edm.String	The passphrase for the restore image.
IN	Password	Edm.String	Password for the remote share.
IN	PreserveVDConfig	Edm.String	Specifies whether to preserve the virtual drive configuration.
IN	ProxyPasswd	Edm.String	The password for the proxy server.
IN	ProxyPort	Edm.String	Port for the proxy server. Default is set to 80.
IN	ProxyServer	Edm.String	The IP address of the proxy server.
IN	ProxySupport	Edm.String	Specifies if proxy should be used. Default is 1 (Off).
IN	ProxyType	Edm.String	The proxy type of the proxy server. Default is 0 (HTTP).
IN	ProxyUname	Edm.String	The user name for the proxy server.
IN	ScheduledStartTime	Edm.String	Schedules the job at the time specified. The format is `yyyymmddhhmmss`. Default setting is `TIME_NOW`, which will start the job immediately.
IN	ShareName	Edm.String	Name of the CIFS share or full path to the NFS share. Optional for HTTP/HTTPS share, which may be treated as the path of the directory containing the file.
IN	ShareType	Edm.String	Type of the network share. Default is NFS, when this parameter is not passed.
IN	UntilTime	Edm.String	End time for the job execution in format: `yyyymmddhhmmss`. If not specified, there is no end time.
IN	UserName	Edm.String	The user name to access the network share.
IN	Workgroup	Edm.String	Optional. Workgroup for the CIFS share.
OUT	Job	Edm.String	Reference to the job spawned if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message corresponding to the MessageID is returned, in English.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.3 DellLicenseManagementService

The DellLicenseManagementService resource provides actions to support License Management functionality.

URI : `/redfish/v1/Dell/Managers/iDRAC.Embedded.1/DellLicenseManagementService`

3.3.1 DeleteLicense

A method used to delete assigned licenses. The DeleteLicense method deletes a specific license from all devices it is assigned to if only the EntitlementID parameter is present, all licenses from a specific device if only the FQDD parameter is present, or a specific license from a specific device if both the EntitlementID and FQDD parameters are present. Either the EntitlementID or FQDD parameter shall be present.

DeleteLicense Method Parameters

Qualifiers	Name	Type	Description
IN, Required	DeleteOptions	Edm.String	Flag used to force delete or delete license from all like devices.
IN	EntitlementID	Edm.String	Entitlement ID of the license to delete.
IN	FQDD	Edm.String	FQDD of the device to delete the license from.
OUT	Message	Edm.String	Error or information message, in English, corresponding to MessageID is returned.
OUT	MessageArgs	Edm.String	Substitution variables for dynamic error messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.3.2 ExportLicense

ExportLicense is a method used to export License files from the iDRAC. The ExportLicense method returns a License, specified by Entitlement ID, as a base64–encoded string.

ExportLicense Method Parameters

Qualifiers	Name	Type	Description
IN, Required	EntitlementID	Edm.String	Entitlement ID of the license being exported.
OUT	LicenseFile	Edm.String	Base-64 encoded string of the license file contents.
OUT	Message	Edm.String	Error or information message, in English, corresponding to MessageID is returned.
OUT	MessageArgs	Edm.String	Substitution variables for dynamic error messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.3.3 ExportLicenseByDeviceToNetworkShare

`ExportLicenseByDeviceToNetworkShare` is a method used to export License files from the IDRAC. The ExportLicenseByDeviceToNetworkShare method exports all licenses from a device, specified by FQDD, to a user-defined location. The following tables specify ExportLicenseByDeviceToNetworkShare return values and parameters.

ExportLicenseByDeviceToNetworkShare Method Parameters

Qualifiers	Name	Type	Description
IN, Required	FQDD	Edm.String	FQDD of the device to export licenses from.
IN, Required	FileName	Edm.String	The exported license is renamed to <*FileName*>.
IN, Required	IPAddress	Edm.String	IP Address of the machine hosting the CIFS, NFS, HTTP, or HTTPS share.
IN	Password	Edm.String	Password for CIFS share authentication.
IN, Required	ShareName	Edm.String	Name of the CIFS share or full file folder path to the NFS share. Optional for HTTP/HTTPS share, which may be treated as the path of the directory containing the file.
IN, Required	ShareType	Edm.String	Type of the network share.
IN	UserName	Edm.String	Username for CIFS share authentication.
IN	Workgroup	Edm.String	Workgroup for the CIFS share, optional.
OUT	Job	Edm.String	Reference to the job spawned, if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message, in English, corresponding to the Message ID is returned.
OUT	MessageArgs	Edm.String	Substitution variables for dynamic error messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.3.4 ExportLicenseToNetworkShare

`ExportLicenseToNetworkShare` is a method used to export License files from the IDRAC. The ExportLicenseToNetworkShare method exports a License, specified by EntitlementID, to a user-defined location.

ExportLicenseToNetworkShare Method Parameters

Qualifiers	Name	Type	Description
IN, Required	EntitlementID	Edm.String	Entitlement ID of the license being exported.
IN	FileName	Edm.String	If included, the exported license is renamed as <*FileName*>.
IN, Required	IPAddress	Edm.String	IP address of the server hosting the network share.
IN	IgnoreCertWarning	Edm.String	Specifies if certificate warning is to be ignored when HTTPS is specified. If IgnoreCertWarning is on, certificate will be ignored. Default is 1 (Off).
IN	Password	Edm.String	Password for Network share authentication.
IN	ProxyPasswd	Edm.String	The password to log in to the proxy server.

IN	ProxyPort	Edm.String	Port for the proxy server. Default is set to 80.
IN	ProxyServer	Edm.String	The IP Address of the proxy server.
IN	ProxySupport	Edm.String	Specifies if proxy is to be used or not. Default is 1 (Off).
IN	ProxyType	Edm.String	The proxy type of the proxy server. Default is 0 (HTTP).
IN	ProxyUname	Edm.String	The user name for proxy server.
IN	ShareName	Edm.String	Name of the CIFS share or full path to the NFS share. Optional for HTTP or HTTPS share, which may be treated as the path of the directory containing the file.
IN, Required	ShareType	Edm.String	Type of the network share.
IN	UserName	Edm.String	Username for the network share authentication.
IN	Workgroup	Edm.String	Optional. Workgroup for the CIFS share.
OUT	Job	Edm.String	Reference to the job spawned, if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message, in English, corresponding to Message ID is returned.
OUT	MessageArgs	Edm.String	Substitution variables for dynamic error messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.3.5 ImportLicense

The ImportLicense method imports the License given in the input parameter to the License Manager Data Store.

ImportLicense Method Parameters

Qualifiers	Name	Type	Description
IN, Required	FQDD	Edm.String	FQDD of the device to apply the license to.
IN, Required	ImportOptions	Edm.String	Flag to force or install for all similar devices.
IN, Required	LicenseFile	Edm.String	A base-64 encoded string of the XML License file.
OUT	Message	Edm.String	Error or information message, in English, corresponding to the Message ID is returned.
OUT	MessageArgs	Edm.String	Substitution variables for dynamic error messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.3.6 ImportLicenseFromNetworkShare

The ImportLicenseFromNetworkShare method imports the License given in the network share location.

ImportLicenseFromNetworkShare Method Parameters

Qualifiers	Name	Type	Description
IN, Required	FQDD	Edm.String	FQDD of the device to apply the license to.
IN, Required	IPAddress	Edm.String	IP address of the server hosting the network share.
IN	IgnoreCertWarning	Edm.String	Specifies if certificate warning is to be ignored when HTTPS is specified. If IgnoreCertWarning is On, the certificate will be ignored. Default is 1 (Off).
IN, Required	ImportOptions	Edm.String	Flag to force or install for all similar devices.
IN, Required	LicenseName	Edm.String	The file name of the license file to be imported.
IN	Password	Edm.String	Password for Network share authentication.
IN	ProxyPasswd	Edm.String	The password to log in proxy server.
IN	ProxyPort	Edm.String	Port for the proxy server. Default is set to 80.
IN	ProxyServer	Edm.String	The IP Address of the proxy server.
IN	ProxySupport	Edm.String	Specifies if proxy is to be used or not. Default is 1 (Off).
IN	ProxyType	Edm.String	The proxy type of the proxy server. Default is 0 (HTTP).
IN	ProxyUname	Edm.String	The username for proxy server.
IN	ShareName	Edm.String	Name of the CIFS share or full path to the NFS share. Optional for HTTP/HTTPS share, this may be treated as the path of the directory containing the file.
IN, Required	ShareType	Edm.String	Type of the network share.
IN	UserName	Edm.String	User name for network share authentication.
IN	Workgroup	Edm.String	Optional. Workgroup for the CIFS share.
OUT	Job	Edm.String	Reference to the job spawned, if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message, in English, corresponding to the message ID is returned.
OUT	MessageArgs	Edm.String	Substitution variables for dynamic error messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.3.7 ShowLicenseBits

The ShowLicenseBits method is used to retrieve the iDRAC feature license bit string for the Licenses in iDRAC in a hexadecimal representation of a 256 bit-string. To read the output, you will need to see which bit position is set to TRUE from right-to-left. Refer to the Appendix section that will show which bit represents which feature.

ShowLicenseBits Method Parameters

Qualifiers	Name	Type	Description
OUT	LicenseBits	Edm.String	The hexadecimal equivalent of the 256-bit license-bit string.
OUT	Message	Edm.String	Error or information message, in English, corresponding to the message ID is returned.
OUT	MessageArgs	Edm.String	Substitution variables for dynamic error messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.4 DellOSDeploymentService

The DellOSDeploymentService resource provides actions to support OS deployment configurations.

URI: `/redfish/v1/Dell/Systems/System.Embedded.1/DellOSDeploymentService.`

3.4.1 BootToHD

The BootToHD method is used for one-time boot to the host server's hard drive.

BootToHD Method Parameters

Qualifiers	Name	Type	Description
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.4.2 BootToISOFromVFlash

The BootToISOFromVFlash method is used to boot to the downloaded pre-operating system image on the vFlash SD card. The following table lists the return values for BootToISOFromVFlash method, where the method-execution behavior matches the return-code description.

BootToISOFromVFlash Method Parameters

Qualifiers	Name	Type	Description
IN	ExposeDuration	Edm.DateTimeOffset	Identifies the amount of time, up to 18 hours, for the drivers to be exposed as a USB device to the host. The default value shall be 18 hours, if the parameter is not specified. The format for intervals is: `YYYY-MM-DDThh:mm:ss[.SSS](Z|(+|-)hh:mm)`.
OUT	Job	Edm.String	Reference to the job spawned, if the operation continues after the method returns.

OUT	Message	Edm.String	Error or information message, in English, corresponding to Message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.4.3 BootToNetworkISO

The BootToNetworkISO method is used to boot from an image file mounted on iDRAC virtual media. The following table lists the return values for the `BootToNetworkISO` method, where the method-execution behavior matches the return-code description.

BootToNetworkISO Method Parameters

Qualifiers	Name	Type	Description
IN	ExposeDuration	Edm.DateTimeOffset	Identifies the amount of time (up to 18 hours) for the ISO image file to be exposed as a local CD-ROM device to the host. The format for intervals is: `YYYY-MM-DDThh:mm:ss[.SSS](Z|(+|-)hh:mm)`
IN	FolderName	Edm.String	Folder name containing the ISO image file.
IN	HashType	Edm.String	Type of hash algorithm used to compute checksum.
IN	HashValue	Edm.String	Checksum value in string format computed by using the 'HashType' algorithm.
IN, Required	IPAddress	Edm.String	NFS, CIFS, HTTP, or HTTPS share IPv4 address. For example, `192.168.10.100`.
IN, Required	ImageName	Edm.String	Name of the image file.
IN	Password	Edm.String	Password, if applicable.
IN, Required	ShareName	Edm.String	Name of the CIFS share or full path to the NFS share. Optional for HTTP or HTTPS share, this may be treated as the path of the directory containing the file.
IN, Required	ShareType	Edm.String	Type of the Network Share.
IN	UserName	Edm.String	User name, if applicable. If the value contains domain name—for example, domain name or user name—the "domain name" will be considered as the work group. Work group parameter will take precedence if value is separately passed for it along with a user name. Supports only '' and '/' for separator.
IN	Workgroup	Edm.String	Optional. Workgroup for the CIFS share.
OUT	Job	Edm.String	Reference to the job spawned, if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message, in English, corresponding to the message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.4.4 ConfigurableBootToNetworkISO

The ConfigurableBootToNetworkISO method exposes an ISO Image present on a network share as a CD-ROM device to the host server for a specified exposure duration interval or by default for 18 hrs. Upon the successful execution, based on the ResetType parameter, the host system shall either immediately cold boot or warm boot. After this reset, the system shall boot to the ISO image file. If ResetType specifies no immediate reboot then after the next host system reset, the system will boot to the ISO Image. Also, if immediate reset is not specified then the system must be rebooted before the exposure duration interval expires. Else, the system will fail to boot to the ISO image file.

ConfigurableBootToNetworkISO Method Parameters

Qualifiers	Name	Type	Description
IN	ExposeDuration	Edm.DateTimeOffset	Identifies the amount of time (up to 18 hours) for the ISO image file to be exposed as a local CD-ROM device to the host server after which it will be automatically detached. The default value shall be 18 hours, if the parameter is not specified. The format for intervals is: `YYYY-MM-DDThh:mm:ss[.SSS](Z|(+|-)hh:mm)`
IN	HashType	Edm.String	Type of hash algorithm used to compute checksum.
IN	HashValue	Edm.String	Checksum value in string format computed by using the 'HashType' algorithm.
IN, Required	IPAddress	Edm.String	NFS, CIFS, HTTP, HTTPS share IPv4 address. For example, `192.168.10.100`.
IN, Required	ImageName	Edm.String	Name of the image file.
IN	Password	Edm.String	Password, if applicable.
IN, Required	ResetType	Edm.String	Specifies if the host system needs to be immediately forced to cold- or warm reset in order to boot to the ISO Image.
IN, Required	ShareName	Edm.String	Name of the CIFS share or full path to the NFS share. Optional for HTTP or HTTPS share, this may be treated as the path of the directory containing the file.
IN, Required	ShareType	Edm.String	Type of the network share.
IN	UserName	Edm.String	User name, if applicable.
IN	Workgroup	Edm.String	Optional. Workgroup for the CIFS share.
OUT	Job	Edm.String	Reference to the job spawned, if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.4.5 ConnectNetworkISOImage

The ConnectNetworkISOImage method is used to connect to the ISO present on the network share and expose the ISO as a local USB CD-ROM device to the host system. This method will connect to the ISO located on an NFS, CIFS, HTTP or HTTPS share and expose it as a virtual CD-ROM device to the host server. Even though the successful method execution will not change the boot order of that device, the host shall always boot to the virtual CD-ROM. Also note that after the ISO is exposed to the host

server, Lifecycle Controller will be locked and no other jobs like configuration or update can be performed until the ISO is detached using the `DisconnectNetworkISOImage` method. The successful execution of the DisconnectNetworkISOImage will revert the host system to the regular boot list.

NOTE—The recommended methodology for connecting to an ISO image is by using the ConnectRFSISOImage method that utilizes the Remote File System (RFS).

ConnectNetworkISOImage Method Parameters

Qualifiers	Name	Type	Description
IN	HashType	Edm.String	Type of hash algorithm used to compute checksum.
IN	HashValue	Edm.String	Checksum value in string format computed by using HashType algorithm.
IN, Required	IPAddress	Edm.String	CIFS, NFS, HTTP, or HTTPS share IPv4 address. For example, `192.168.10.100`.
IN, Required	ImageName	Edm.String	Name of the image file.
IN	Password	Edm.String	Password of the account to access the share.
IN, Required	ShareName	Edm.String	Name of the CIFS share or full path to the NFS share. Optional for HTTP or HTTPS share, which may be treated as the path of the directory containing the file.
IN, Required	ShareType	Edm.String	Type of the network share.
IN	UserName	Edm.String	User name of the account to access the share.
IN	Workgroup	Edm.String	Optional. Workgroup for the CIFS share.
OUT	Job	Edm.String	Reference to the job spawned, if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message, in English, corresponding to the message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.4.6 DeleteISOFromVFlash

The DeleteISOFromVFlash method is used to delete the ISO Image from vFlash SD card. The following table lists the return values for the DeleteISOFromVFlash method, where the method- execution behavior matches the return-code description.

DeleteISOFromVFlash Method Parameters

Qualifiers	Name	Type	Description
OUT	Message	Edm.String	Error or information message, in English, corresponding to MessageID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.4.7 DetachDrivers

This `OSDeploymentService.DetachDrivers` method is used to detach the USB device containing the Lifecycle Controller driver pack from the host server. The following table lists the return values for DetachDrivers method, where the method-execution behavior matches the return-code description.

DetachDrivers Method Parameters

Qualifiers	Name	Type	Description
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.4.8 DetachISOFromVFlash

The DetachISOFromVFlash method is used to detach the ISO Image on a vFlash SD card from the host system.

DetachISOFromVFlash Method Parameters

Qualifiers	Name	Type	Description
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.4.9 DetachISOImage

The DetachISOImage method is used to detach an ISO Image from the host server. The following table lists the return values for DetachISOImage method, where the method-execution behavior matches the return-code description. The following table lists the return values for BootToNetworkISO method, where the method-execution behavior matches the return-code description.

DetachISOImage Method Parameters

Qualifiers	Name	Type	Description
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.4.10 DisconnectNetworkISOImage

The DisconnectNetworkISOImage method is used to disconnect and detach the ISO Image from the host system. The following table lists the return values for the DisconnectNetworkISOImage method, where the method-execution behavior matches the return-code description.

DisconnectNetworkISOImage Method Parameters

Qualifiers	Name	Type	Description
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.4.11 DownloadISOToVFlash

The DownloadISOToVFlash method is used to download the pre-operating system ISO Image to the vFlash SD card. The following table lists the return values for the `DownloadISOToVFlash` method, where the method-execution behavior matches the return-code description.

DownloadISOToVFlash Method Parameters

Qualifiers	Name	Type	Description
IN	HashType	Edm.String	Type of hash algorithm used to compute checksum.
IN	HashValue	Edm.String	Checksum value in string format computed using 'HashType' algorithm.
IN, Required	IPAddress	Edm.String	The TFTP, CIFS, NFS, HTTP, or HTTPS share IPv4 address. For example, `192.168.10.100`.
IN, Required	ImageName	Edm.String	Name of the image file.
IN	Password	Edm.String	Password, if applicable.
IN	Port	Edm.Int64	Port Number
IN, Required	ShareName	Edm.String	Name of the CIFS share or full path to the NFS share. Optional for HTTP/HTTPS share, which may be treated as the path of the directory containing the file.
IN, Required	ShareType	Edm.String	Type of the network share.
IN	UserName	Edm.String	User name, if applicable.
IN	Workgroup	Edm.String	Optinal. Workgroup for the CIFS share.
OUT	Job	Edm.String	Reference to the job spawned, if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.4.12 GetAttachStatus

The GetAttachStatus method will give the status of the Lifecycle Controller Driver Pack and ISO Image that may be attached to the host.

GetAttachStatus Method Parameters

Qualifiers	Name	Type	Description
OUT	DriversAttachStatus	Edm.String	Indicates if the driver is attached.
OUT	ISOAttachStatus	Edm.String	Indicates if the ISO image file is attached.
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.4.13 GetDriverPackInfo

The GetDriverPackInfo method is used get the list of operating systems by the Lifecycle Controller Driver Pack.

GetDriverPackInfo Method Parameters

Qualifiers	Name	Type	Description
OUT	Job	Edm.String	Reference to the job spawned, if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	OSList	Edm.String	NULL or the list of OSs supported by this server.
OUT	Version	Edm.String	NULL or version of the driver pack.

3.4.14 GetNetworkISOImageConnectionInfo

The GetNetworkISOImageConnectionInfo method is used to give the status of the ISO Image file that has been exposed to the host system. The following table lists the return values for GetNetworkISOImageConnectionInfo, where the method-execution behavior matches the return-code description.

GetNetworkISOImageConnectionInfo Method Parameters

Qualifiers	Name	Type	Description
OUT	HostAttachedStatus	Edm.String	Indicates whether or not the ISO image file is attached to the host server.
OUT	HostBootedFromISO	Edm.String	Indicates whether or not the host has booted to the ISO image at least once.
OUT	IPAddr	Edm.String	IPv4 address of the CIFS, NFS, HTTP, or HTTPS share. Example `192.168.10.100`.
OUT	ISOConnectionStatus	Edm.String	Indicates whether or not the ISO file is still accessible.
OUT	ImageName	Edm.String	ISO image name as specified in the the `ConnectNetworkISOImage` method.
OUT	Message	Edm.String	Error or information message, in English, corresponding to the message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	ShareName	Edm.String	Share name as specified in the the `ConnectNetworkISOImage` method.
OUT	UserName	Edm.String	User name as specified in the `ConnectNetworkISOImage` method.
OUT	Workgroup	Edm.String	Workgroup as specified in the `ConnectNetworkISOImage` method.

3.4.15 UnpackAndAttach

The UnpackAndAttach method is used to extract the Lifecycle Controller driver pack for the selected OS to a USB device that is attached locally to the server for the specified time interval. The following table specifies the return values for the `UnpackAndAttach` method, where the method-execution behavior matches the return-code description.

UnpackAndAttach Method Parameters

Qualifiers	Name	Type	Description
IN	ExposeDuration	Edm.DateTimeOffset	Identifies the amount of time (up to 18 hours) for the drivers to be exposed as an USB device to the host. The default value shall be 18 hours, if the parameter is not specified. The format for intervals is: `YYYY-MM-DDThh:mm:ss[.SSS](Z|(+|-)hh:mm)`
IN, Required	OSName	Edm.String	Name of the OS to unpack drivers for, this value shall match one of the strings in OSList returned for GetDriverPackInfo.
OUT	Job	Edm.String	Reference to the job spawned, if the operation continues after the method returns.

OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.4.16 UnpackAndShare

The UnpackAndShare method is used to extract the Lifecycle Controller Driver Pack for the selected operating system and copy them to the specified network share. The following table lists the return values for UnpackAndShare, where the method-execution behavior matches the return-code description.

UnpackAndShare Method Parameters

Qualifiers	Name	Type	Description
IN	FolderName	Edm.String	Folder name.
IN, Required	IPAddress	Edm.String	CIFS, NFS, HTTP, or HTTPS share IPv4 address. For example, `192.168.10.100`.
IN, Required	OSName	Edm.String	The OS name.
IN	Password	Edm.String	Password, if applicable.
IN, Required	ShareName	Edm.String	Name of the CIFS share or full path to the NFS share. Optional for HTTP/HTTPS share, which may be treated as the path of the directory containing the file.
IN, Required	ShareType	Edm.String	Type of the network share.
IN	UserName	Edm.String	User name, if applicable.
IN	Workgroup	Edm.String	Optional. Workgroup for the CIFS share.
OUT	Job	Edm.String	Reference to the job spawned, if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.4.17 OS Deployment workflow by using the BootToNetworkISO method

In this OS deployment workflow example, the Lifecycle Controller driver pack will be unpacked and attached, and the server booted to a network ISO attached from a network share to perform an operating system installation. The `UnpackAndAttachOsdREDFISH.py` and `BootTonetworkIsoREDFISH.py` Python scripts are used.

1. Check which OS drivers are available from the Lifecycle Controller driver pack:

```
C:\Python27>UnpackAndAttachOsdREDFISH.py -ip 192.168.0.120 -u root -p calvin
-g y
- PASS: POST command passed to get driver pack information, status code 200
returned
- Driver packs supported for iDRAC 100.65.205.134
Microsoft Windows Server 2016
```

```
Microsoft Windows Server 2012 R2
Microsoft Windows Server 2019
Red Hat Enterprise Linux 6.10 x64
Red Hat Enterprise Linux 7.5 x64
SuSE Enterprise Linux 15 x64
```

2. Unpack and attach the driver pack for the OS to be installed; this workflow will install Microsoft Windows Server 2012:

```
C:\Python27>UnpackAndAttachOsdREDFISH.py -ip 192.168.0.120 -u root -p calvin
-U "Microsoft Windows Server 2012 R2"
- PASS: POST command passed for UnpackAndAttach method, status code 202
returned
- WARNING, concrete job URI created for method UnpackAndAttach:
/redfish/v1/TaskService/Tasks/OSDeployment

- WARNING, concrete job not completed, current status is: "Running", job
execution time is "0:00:00"
- WARNING, concrete job not completed, current status is: "Running", job
execution time is "0:00:04"
- WARNING, concrete job not completed, current status is: "Running", job
execution time is "0:00:07"
- PASS, concrete job successfully marked completed
- Final detailed job results -
Description: Server Configuration and other Tasks running on iDRAC are
listed here
TaskState: Completed
Message: The command was successful.
MessageId: OSD1
MessageArgs: []
MessageArgs@odata.count: 0
@odata.id: /redfish/v1/TaskService/Tasks/OSDeployment
@odata.context: /redfish/v1/$metadata#Task.Task
TaskStatus: OK
Messages@odata.count: 1
StartTime: TIME_NOW
EndTime:
Id: OSDeployment
Name: UnpackAndAttach

- Concrete job completed in 0:00:33

- PASS, driver pack attach status successfully identified as "Attached"
```

3. Boot the server from an OS ISO image file which is located on a network shared location. After executing the script to boot to the network ISO, the server will immediately reboot, a flag will be set in POST to boot to the ISO and after the POST completes, OS installation will begin. If the OS ISO is not pre-configured for unattended install, manual interaction with the OS installation may be required to start the process:

```
C:\Python27>BootToNetworkIsoOsdREDFISH.py -ip 192.168.0.120 -u root -p
calvin -b y --ipaddress 192.168.0.130 --sharetype NFS --sharename
nfs_share_vm --imagename WS2012R2.ISO

- WARNING, arguments and values used to BootToNetworkISO on network share

ShareType: NFS
ShareName: nfs_share_vm
ImageName: WS2012R2.ISO
IPAddress: 192.168.0.130

- PASS: POST command passed for BootToNetworkISO method, status code 202
returned
- WARNING, concrete job URI created for method BootToNetworkISO:
/redfish/v1/TaskService/Tasks/OSDeployment

- WARNING, concrete job not completed, current status is: "Running", job
execution time is "0:00:00"
- WARNING, concrete job not completed, current status is: "Running", job
execution time is "0:00:06"
- WARNING, concrete job not completed, current status is: "Running", job
execution time is "0:00:12"
- WARNING, concrete job not completed, current status is: "Running", job
execution time is "0:00:17"
- PASS, concrete job successfully marked completed

- Final detailed job results -

Description: Server Configuration and other Tasks running on iDRAC are
listed here
TaskState: Completed
Message: The command was successful.
MessageId: OSD1
MessageArgs: []
MessageArgs@odata.count: 0
@odata.id: /redfish/v1/TaskService/Tasks/OSDeployment
@odata.context: /redfish/v1/$metadata#Task.Task
TaskStatus: OK
Messages@odata.count: 1
StartTime: TIME_NOW
EndTime:
Id: OSDeployment
Name: BootToNetworkISO

- concrete job completed in 0:01:29
- PASS, ISO attach status successfully identified as "Attached"
```

4. After the OS installation has completed, the ISO image and driver pack will remain attached for 18 hours and then be automatically unattached. If the ISO or driver pack are no longer needed, the ISO image and Driver Pack can be manually detached using the scripts:

```
C:\Python27>UnpackAndAttachOsdREDFISH.py -ip 192.168.0.120 -u root -p calvin
-d y

- PASS: POST command passed to detach driver pack, status code 200 returned
- PASS, driver pack attach status successfully identified as "NotAttached"

C:\Python27>BootToNetworkIsoOsdREDFISH.py -ip 192.168.0.120 -u root -p
calvin -d y

- PASS: POST command passed to detach ISO image, status code 200 returned
- PASS, ISO attach status successfully identified as "NotAttached"
```

3.5 DellPersistentStorageService

The DellPersistentStorageService resource provides actions to support vFlash functionality.

URI: `/redfish/v1/Dell/Managers/iDRAC.Embedded.1/DellPersistentStorageService`

3.5.1 FormatPartition

The FormatPartition Action is used for formatting a partition on a vFlash.

FormatPartition Method Parameters

Qualifiers	Name	Type	Description
IN, Required	FormatType	Edm.String	The list of types to format to.
IN, Required	PartitionIndex	Edm.Int64	The index of the partition that shall have value be between 1 and 16.
OUT	Job	Edm.String	Reference to the job spawned if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.6 DellRAIDService

The DellRaidService resource provides actions to support PERC RAID controller functionality.

URI: `/redfish/v1/Dell/Systems/System.Embedded.1/DellRaidService`

3.6.1 AssignSpare

The AssignSpare method is used to assign a physical disk as a dedicated hot spare for a virtual disk, or as a global hot spare.

AssignSpare Method Parameters

Qualifiers	Name	Type	Description
IN, Required	Target	Edm.String	This pParameter is the FQDD of the target device (physical drive).
IN	VirtualDiskArray	Edm.String	Array of ElementName(s) where each identifies a different Virtual Drive (VD). Currently only one VD can be passed. Array of ElementName(s) where each ElementName identifies a different virtual disk.
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	RebootRequired	Edm.String	Indicates whether a reboot is required to complete the operation performed. The value is one of the following: • Yes—Reboot is required to perform the operation • No—Reboot is not required to perform the operation • Optional—The operation can be performed with or without reboot based on the type of job created

3.6.2 BlinkTarget

The BlinkTarget method is used to identify a single physical drive by blinking the drive slot LED for the physical disk / Virtual Disk. The successful execution of this method results in setting the LED to blink the identify pattern or turns off the blinking of the identify pattern. The method is real time; blink cannot be scheduled as part of a job.

BlinkTarget Method Parameters

Qualifiers	Name	Type	Description
IN, Required	Target	Edm.String	This parameter is the FQDD of the physical drive, SSD, or VD.
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	List of message IDs for the output message.
OUT	RebootRequired	Edm.String	Indicates whether a reboot is required to complete the operation performed. The value is one of the following: • Yes—Reboot is required to perform the operation • No—Reboot is not required to perform the operation • Optional—The operation can be performed with or without reboot based on the type of job created.

3.6.3 CheckVDValues

The CheckVDValues method is used to determine the possible sizes of VDs and the default settings, based upon a RAID level and set of physical disks. The VDPropArray property is filled with Size and other values, so that the method is successfully executed. If the SpanDepth is not provided, a default value of 2 shall be used for RAID levels 10, 50, and 60. NOTE: For certain numbers of disks such as nine or fifteen, it may be necessary for the user to provide another SpanDepth.

CheckVDValues Method Parameters

Qualifiers	Name	Type	Description
IN, Required	PDArray	Edm.String	Array of FQDDs where each identifies a physical drive.
IN, Required	Target	Edm.String	FQDD of the target device (Controller).
IN, Required	VDPropNameArrayIn	Edm.String	Indexed array of VD property names. The property names can be Size, SpanDepth, SpanLength, RAIDLevel, StartingLBA or T10PIStatus. The values for the property needs to be provided in `VDPropValueArrayIn` in the same order.
IN, Required	VDPropValueArrayIn	Edm.String	Indexed array of VD property values for the respective `VDPropNameArrayIn` parameter.
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	VDPropNameArray	Edm.String	Indexed array of the Virtual Disk property names with relative values contained in the `VDPropValueArray` parameter.
OUT	VDPropValueArray	Edm.String	Indexed array of VD property values relative to the `VDPropValueName` parameter.

3.6.4 ClearForeignConfig

The ClearForeignConfig method is used to prepare any foreign physical drives for inclusion in the local RAID configuration.

ClearForeignConfig Method Parameters

Qualifiers	Name	Type	Description
IN, Required	Target	Edm.String	FQDD of the target device (Controller).
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	RebootRequired	Edm.String	Indicates whether a reboot is required to complete the operation performed. The value is one of the following: • Yes—Reboot required to perform the operation • No—Reboot not required to perform the operation • Optional—The operation can be performed with or without reboot based on the type of job created

3.6.5 ConvertToNonRAID

The ConvertToNonRAID method is used to convert physical disks in RAID state of "Ready" to a Non-RAID state. After the method is successfully executed, the Dell_PhysicalDiskView.RAIDStatus property of that physical disk should reflect the new state.

ConvertToNonRAID Method Parameters

Qualifiers	Name	Type	Description
IN, Required	PDArray	Edm.String	An array of FQDDs where each identifies a physical drive.
OUT	Message	Edm.String	Error or information message, in English, corresponding to the message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	List of message ID for the output message.
OUT	RebootRequired	Edm.String	Indicates whether a reboot is required to complete the operation performed. The value is one of the following: • Yes—Reboot required to perform the operation • No—Reboot not required to perform the operation • Optional—The operation can be performed with or without reboot based on the type of job created

3.6.6 ConvertToRAID

The ConvertToRAID method is used to convert physical disks in Non-RAID state to a state usable for RAID. After the method is successfully executed the Dell_PhysicalDiskView.RAIDStatus property of that physical disk should reflect the new state.

ConvertToRAID Method Parameters

Qualifiers	Name	Type	Description
IN, Required	PDArray	Edm.String	An array of FQDDs where each identifies a physical drive.
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	List of message IDs for the output message.
OUT	RebootRequired	Edm.String	Indicates whether a reboot is required to complete the operation performed. The value is one of the following: • Yes—Reboot required to perform the operation • No—Reboot not required to perform the operation • Optional—The operation can be performed with or without reboot based on the type of job created

3.6.7 EnableControllerEncryption

The EnableControllerEncryption method sets Local Key Management (LKM) on controllers that support encryption of the drives.

EnableControllerEncryption Method Parameters

Qualifiers	Name	Type	Description
IN	Key	Edm.String	Key is the passcode. This parameter is required if the mode is set to Local Key Management. The key shall be maximum of 32 characters in length, where the expanded form of the special character is counted as a single character. The Key shall have one character from each of the following set—Upper case, lower case, number, special character. The special characters in the following set must be passed as mentioned below. `&->&,< -><, >->>, "->&quot;,` and `'->&apos;"`.
IN	Keyid	Edm.String	Key identifier describes the key. This parameter is required if the mode is set to Local Key Management. The key ID shall be maximum of 32 characters in length and must not have any white spaces.
IN, Required	Mode	Edm.String	Mode of the controller 1 - Local Key Management (LKM).
IN, Required	Target	Edm.String	FQDD of target device (Controller).
OUT	Message	Edm.String	Error or information message, in English, corresponding to the message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	RebootRequired	Edm.String	Indicates whether a reboot is required to complete the operation performed. The value is one of the following: • Yes—Reboot required to perform the operation • No—Reboot not required to perform the operation • Optional—The operation can be performed with or without reboot based on the type of job created

3.6.8 GetAvailableDisks

The GetAvailableDisks method is used to determine minimum number of physical drives required to create virtual disks.

GetAvailableDisks Method Parameters

Qualifiers	Name	Type	Description
IN	BlockSizeInBytes	Edm.String	The parameter specifies the physical drive block size in bytes.
IN	DiskEncrypt	Edm.String	The parameter specifies the drive encryption capability.
IN, Required	DiskType	Edm.String	The property represents the drive type.
IN, Required	Diskprotocol	Edm.String	The parameter specifies the type of drive protocol.
IN	FormFactor	Edm.String	This parameter is used to specify the form factor of drives and shall be one of the following: • 0—Include all

			• 1—Include only BOSS M.2
IN	RaidLevel	Edm.String	The parameter represents the type of RAID configuration.
IN	T10PIStatus	Edm.String	The parameter specifies the T10-Protection Information (PI) capability status.
IN, Required	Target	Edm.String	This parameter is the FQDD of the target device (Controller).
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	PDArray	Edm.String	An array of FQDD(s) identifies physical drive (s).

3.6.9 GetDHSDisks

The GetDHSDisks method is used to determine possible choices of physical drives that can used to set a dedicated hot-spare for the identified VD. GetDHSDisks returns success if it has evaluated the physical drives for potential hot-spares, the PDArray return list can be empty if no physical drives are suitable for hot-spares.

GetDHSDisks Method Parameters

Qualifiers	Name	Type	Description
IN, Required	Target	Edm.String	This parameter is the FQDD of the target device (VD).
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	PDArray	Edm.String	An array of FQDDs where each entry identifies a physical drive.

3.6.10 GetRAIDLevels

The GetRAIDLevels method is used to determine the possible choices of RAID Levels to create virtual disks. If the list of physical disks is not provided, this method accesses information for all the connected disks.

GetRAIDLevels Method Parameters

Qualifiers	Name	Type	Description
IN	BlockSizeInBytes	Edm.String	The parameter specifies the physical drive block size in bytes.
IN	DiskEncrypt	Edm.String	The parameter specifies the drive encryption capability.
IN, Required	DiskType	Edm.String	The parameter specifies the type of the drive.
IN, Required	Diskprotocol	Edm.String	The parameter specifies the type of drive protocol.
IN	FormFactor	Edm.String	This parameter is used to specify the form factor of drives: • 0—Include all • 1—Include only BOSS M.

IN	PDArray	Edm.String	Array of FQDDs where each identifies a physical drive.
IN	T10PIStatus	Edm.String	The parameter specifies the T10-Protection Information (PI) capability status.
IN, Required	Target	Edm.String	FQDD of target device (Controller).
OUT	Message	Edm.String	Error or information message, in English, corresponding to MessageID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	VDRAIDEnumArray	Edm.String	Indexed array of VD RAID level enum values.

3.6.11 LockVirtualDisk

The LockVirtualDisk method encrypts the virtual disk.

LockVirtualDisk Method Parameters

Qualifiers	Name	Type	Description
IN, Required	Target	Edm.String	FQDD of target device (VD).
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	RebootRequired	Edm.String	Indicates whether a reboot is required to complete the operation performed. The value is one of the following: • Yes—Reboot required to perform the operation • No—Reboot not required to perform the operation • Optional—The operation can be performed with or without reboot based on the type of job created

3.6.12 ReKey

The ReKey method resets the key on the PERC controller that support encryption of the of drives. This method switches the controller mode.

ReKey Method Parameters

Qualifiers	Name	Type	Description
IN	Keyid	Edm.String	Key identifier describes the key. The key ID shall be maximum 32 characters in length and must not have any white spaces.
IN, Required	Mode	Edm.String	Mode of the controller: 1 - Local Key Management (LKM)
IN, Required	NewKey	Edm.String	New controller key. The key shall be maximum of 32 characters in length, where the expanded form of the special character is counted as a single character. The Key shall have one character from each of the following set. Upper case, lower case, number, special character. The special characters in the following set need to be passed as mentioned below and are counted as a single character for the maximum length of the key. &->&, < -><, >->>, "->", and '->'".

IN, Required	OldKey	Edm.String	The old controller key.
IN, Required	Target	Edm.String	FQDD of target device (Controller).
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	RebootRequired	Edm.String	Indicates whether a reboot is required to complete the operation performed. The value is one of the following: • Yes—Reboot required to perform the operation • No—Reboot not required to perform the operation • Optional—The operation can be performed with or without reboot based on the type of job created

3.6.13 RemoveControllerKey

The `RemoveControllerKey` method erases the encryption key on controller.

Caution—All encrypted drives will be erased by this action and any available data is permanently deleted.

RemoveControllerKey Method Parameters

Qualifiers	Name	Type	Description
IN, Required	Target	Edm.String	FQDD of target device (Controller).
OUT	Message	Edm.String	Error or information message, in English, corresponding to the message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	RebootRequired	Edm.String	Indicates whether a reboot is required to complete the operation performed. The value is one of the following: • Yes—Reboot required to perform the operation • No—Reboot not required to perform the operation • Optional—The operation can be performed with or without reboot based on the type of job created

3.6.14 ResetConfig

The ResetConfig method is used to delete all the VDs and unassign all hot-spare physical drives.

Caution—All data on the existing VDs will be lost.

ResetConfig Method Parameters

Qualifiers	Name	Type	Description
IN, Required	Target	Edm.String	FQDD of target device (Controller).
OUT	Message	Edm.String	Error or information message, in English, corresponding to the message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	RebootRequired	Edm.String	Indicates whether a reboot is required to complete the operation performed. The value is one of the following: • Yes—Reboot required to perform the operation • No—Reboot not required to perform the operation • Optional—The operation can be performed with or without reboot based on the type of job created

3.6.15 SetControllerKey

The SetControllerKey method is used to set the key on controllers and set the controller in Local Key Management (LKM) to encrypt the drives.

SetControllerKey Method Parameters

Qualifiers	Name	Type	Description
IN, Required	Key	Edm.String	The parameter specifies the key passcode. The Key shall be maximum of 32 characters in length, where the expanded form of the special character is counted as a single character. The Key shall have at least one character from each of the following sets. Upper case, lower case, number, special character, The special characters in the following set need to be passed as mentioned below: &->&, < -><, >->>, "->", ' ->'".
IN, Required	Keyid	Edm.String	Key Identifier that describes the key. The Key ID shall be maximum of 32 characters in length and should not have any spaces.
IN, Required	Target	Edm.String	FQDD of the target device (Controller).
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	RebootRequired	Edm.String	Indicates whether a reboot is required to complete the operation performed. The value is one of the following: • Yes—Reboot required to perform the operation • No—Reboot not required to perform the operation

			• Optional—The operation can be performed with or without reboot based on the type of job created

3.6.16 UnBlinkTarget

The UnblinkTarget method is used to stop blinking the light present on the physical drive represented by the Target FQDD. The method is real time; unblink cannot be scheduled as part of a job.

UnBlinkTarget Method Parameters

Qualifiers	Name	Type	Description
IN, Required	Target	Edm.String	This parameter is the FQDD of the physical drive, SSD, and VD.
OUT	Message	Edm.String	Error or information message, in English, corresponding to message ID is returned.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	RebootRequired	Edm.String	Indicates whether a reboot is required to complete the operation performed. The value is one of the following: • Yes—Reboot required to perform the operation • No—Reboot not required to perform the operation • Optional—The operation can be performed with or without reboot based on the type of job created

3.7 DellSoftwareInstallationService

The DellSoftwareInstallationService resource provides actions to support software installation functionality.

URI :

`/redfish/v1/Dell/Systems/System.Embedded.1/DellSoftwareInstallationService`

3.7.1 GetRepoBasedUpdateList

This method is used for getting the list of packages and a list of devices that will be updated when a repository is used. You must run the InstallFromRespository command before running `GetRepoBasedUpdateList`. If ApplyUpdate is False, no updates are applied or scheduled. If ApplyUpdate is True, the list contains job IDs for all the jobs queued for the devices.

GetRepoBasedUpdateList Method Parameters

Qualifiers	Name	Type	Description
OUT	Message	Edm.String	Error or information message corresponding to the message ID is returned, in English.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.
OUT	PackageList	Edm.String	This property provides all the packages and devices that will be updated when a particular repository is used.

3.7.2 InstallFromRepository

This method creates the list of the updates contained in the repository that are supported by the Lifecycle Controller and are applicable to the system and installed hardware. The successful execution of this method creates a job. The list is copied to a persistent location so that you can query for the updatable list by using GetRepoBasedUpdateList command. If the input parameter ApplyUpdate value is True, only the applicable packages are applied and GetRepoBasedUpdateList gives the JOB IDs for all the jobs queued for the devices.

InstallFromRepository Method Parameters

Qualifiers	Name	Type	Description
IN	ApplyUpdate	Edm.String	If ApplyUpdate is set to True, the updatable packages from Catalog XML are staged. If it is set to False, no updates are applied. The list of updatable packages can be seen by invoking the `GetRepoBasedUpdateList`. Default value is True.
IN	CatalogFile	Edm.String	Name of the catalog file on the repository. Default is Catalog.xml.
IN	IPAddress	Edm.String	IP address for the remote share.
IN	IgnoreCertWarning	Edm.String	Specifies if certificate warning should be ignored when HTTPS is used. If IgnoreCertWarning is On, warnings are ignored. Default is 2 (On).
IN	Mountpoint	Edm.String	The local directory where the share should be mounted. This is applicable for CIFS.
IN	Password	Edm.String	Password for the remote share. This parameter must be provided for CIFS.
IN	ProxyPasswd	Edm.String	The password for the proxy server.
IN	ProxyPort	Edm.Int64	Port for the proxy server. Default is set to 80.
IN	ProxyServer	Edm.String	The IP address of the proxy server.
IN	ProxySupport	Edm.String	Specifies if a proxy should be used. Default is 1 (Off).
IN	ProxyType	Edm.String	The proxy type of the proxy server. Default is 0 (HTTP).
IN	ProxyUname	Edm.String	The user name for the proxy server.
IN	RebootNeeded	Edm.Boolean	This property indicates if a reboot should be performed. True indicates that the system (host) is rebooted during the update process. False indicates that the updates take effect after the system is rebooted the next time. Default value is set to False.
IN	ShareName	Edm.String	Name of the CIFS share or full path to the NFS share. Optional for HTTP/HTTPS share, this may be treated as the path of the directory containing the file.
IN	ShareType	Edm.String	Type of the network share. Default value is NFS.
IN	UserName	Edm.String	User name for the remote share. This parameter must be provided for CIFS.
IN	Workgroup	Edm.String	Workgroup for the CIFS share - optional.
OUT	Job	Edm.String	Reference to the job spawned if the operation continues after the method returns.

OUT	Message	Edm.String	Error or information message corresponding to the MessageID is returned, in English.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.7.3 Firmware Update using custom repository created by using Dell Repository Manager (DRM)

In addition to the standard Redfish APIs for firmware update, a Dell OEM extension API enables update by using a firmware repository created with the Dell Repository Manager (DRM) utility. The Install from Repository API compares the firmware versions contained in the referenced firmware repository and updates the server firmware to match the versions stored in the repository. This enables overall server firmware compliance by using a single API call.

The below example uses the script `InstallFromRepositoryREDFISH.py` to illustrate such an update process.

1. Verify the current firmware versions for all devices in the server the iDRAC supports for updates.

```
C:\Python27>InstallFromRepositoryREDFISH.py -ip 192.168.0.120 -u root -p
calvin -g y
- WARNING, current devices detected with firmware version and updateable
status -
Device Name: PCIe SSD in Slot 2 in Bay 1, Firmware Version: 1.0.0,
Updatable: True
Device Name: BOSS-S1, Firmware Version: 2.5.13.3016, Updatable: True
Device Name: PERC H330 Mini, Firmware Version: 25.5.4.0006, Updatable: True
Device Name: OS Collector, Firmware Version: 0, Updatable: True
Device Name: PCIe SSD in Slot 2, Firmware Version: KPYABD3Q, Updatable: True
Device Name: BP14G+EXP 0:1, Firmware Version: 2.40, Updatable: True
Device Name: iDRAC Service Module Installer, Firmware Version: 0, Updatable:
True
Device Name: Power Supply.Slot.1, Firmware Version: 00.23.32, Updatable:
True
Device Name: Disk 0 on AHCI Controller in slot 1, Firmware Version: DL43,
Updatable: True
Device Name: BIOS, Firmware Version: 2.2.0, Updatable: True
Device Name: OS Drivers Pack, Firmware Version: 0, Updatable: True
Device Name: Disk 1 in Backplane 1 of Integrated RAID Controller 1, Firmware
Version: K774, Updatable: True
Device Name: Integrated Dell Remote Access Controller, Firmware Version:
3.30.30.30, Updatable: True
Device Name: Dell 64 Bit uEFI Diagnostics, version 4301, 4301A18, 4301.19,
Firmware Version: 4301A18, Updatable: True
Device Name: Disk 0 in Backplane 1 of Integrated RAID Controller 1, Firmware
Version: FSF9, Updatable: True
Device Name: QLogic 577xx/578xx 10 Gb Ethernet BCM57800 - 18:66:DA:8E:28:24,
Firmware Version: 08.07.00, Updatable: True
Device Name: System CPLD, Firmware Version: 1.0.2, Updatable: True
```

```
Device Name: Lifecycle Controller, Firmware Version: 3.30.30.30, Updatable:
False
```

For this example, a custom repository has been created and placed on an NFS share. Notice that the NFS share contains a `Catalog.xml` file and the firmware DUPs for the devices to be updated. In this workflow example, the goal is to update the BIOS, PERC controller firmware and the server DIAGs.

```
[root@linux nfs]# ls -la R640_repo_updates/
total 44308
drwxr-xr-x  2 root root     4096 Feb 28 11:53 .
drwxrwxrwx 20 root root    32768 Feb 28 11:53 ..
-rwxr--r--  1 root root 27782392 Feb 28 10:56 BIOS_YJXXX_WN64_1.6.13.EXE
-rwxr--r--  1 root root    24950 Feb 28 11:10 Catalog.xml
-rwxr--r--  1 root root  8483584 Feb 28 10:57
                                               Diagnostics_Application_30H00_
                                               WN64_4301A25_4301.26_01.EXE
-rwxr--r--  1 root root  8967104 Feb 28 10:56 SAS-
                                               RAID_Firmware_F675Y_WN64_25.5.
                                               5.0005_A13_01.EXE
[root@linux nfs]#
```

2. Validate which devices can be updated from the repository. This step is optional, but it is recommended to run this check of the devices that are updateable and to verify the firmware versions to which you wish to update:

```
C:\Python27>InstallFromRepositoryREDFISH.py -ip 192.168.0.120 -u root -p
calvin -i y --ipaddress 192.168.0.130 --sharetype NFS --sharename
/nfs/R640_repo_updates --applyupdate False
- WARNING, arguments and values for InstallFromRepository method
ShareType: NFS
IPAddress: 100.65.84.150
ApplyUpdate: False
ShareName: /nfs/R640_repo_updates

- PASS: POST command passed for method "InstallFromRepository", status code
202 returned
- PASS, job ID JID_513760436205 successfully created
- WARNING, Job ID JID_513760436205 not marked completed, current status:
"Downloading the Catalog.xml update package.", job polling time: "0:00:00"

- PASS, job ID JID_513760436205 successfully marked completed

- Final detailed job results -

@odata.type: #DellJob.v1_0_1.DellJob
JobState: Completed
Description: Job Instance
TargetSettingsURI: None
@odata.id: /redfish/v1/Managers/iDRAC.Embedded.1/Jobs/JID_513760436205
```

```
@odata.context: /redfish/v1/$metadata#DellJob.DellJob
MessageArgs: [u'NA']
CompletionTime: 2019-02-28T11:47:30
PercentComplete: 100
StartTime: TIME_NOW
MessageId: RED001
Message: Job completed successfully.
EndTime: None
Id: JID_513760436205
JobType: RepositoryUpdate
Name: Repository Update

- WARNING, "ApplyUpdate = False" selected, execute script with -r argument
to view the repo update list which will report devices detected for firmware
updates

C:\Python27>InstallFromRepositoryREDFISH.py -ip 192.168.0.120 -u root -p
calvin -r y

-PASS: POST command passed to get repo update list, status code 200 returned

- Repo Based Update List in XML format

<?xml version="1.0"?>
<CIM xmlns:fo="http://www.w3.org/1999/XSL/Format" CIMVERSION="2.0"
DTDVERSION="2.0">
  <MESSAGE ID="4711" PROTOCOLVERSION="1.0">
    <SIMPLEREQ>
      <VALUE.NAMEDINSTANCE>
        <INSTANCENAME CLASSNAME="DCIM_RepoUpdateSWID"><PROPERTY
NAME="Criticality" TYPE="string"><VALUE>3</VALUE></PROPERTY><PROPERTY
NAME="DisplayName" TYPE="string"><VALUE>Dell 64 Bit uEFI Diagnostics,
version 4301, 4301A25, 4301.26</VALUE></PROPERTY><PROPERTY
NAME="BaseLocation" TYPE="string"><VALUE/></PROPERTY><PROPERTY
NAME="PackagePath"
TYPE="string"><VALUE>Diagnostics_Application_30H00_WN64_4301A25_4301.26_01.E
XE</VALUE></PROPERTY><PROPERTY NAME="PackageName"
TYPE="string"><VALUE>Diagnostics_Application_30H00_WN64_4301A25_4301.26_01.E
XE</VALUE></PROPERTY><PROPERTY NAME="PackageVersion"
TYPE="string"><VALUE>4301A25</VALUE></PROPERTY><PROPERTY NAME="RebootType"
TYPE="string"><VALUE>NONE</VALUE></PROPERTY><PROPERTY NAME="JobID"
TYPE="string"><VALUE/></PROPERTY>
    <PROPERTY NAME="Target"
TYPE="string"><VALUE>DCIM:INSTALLED#802__Diagnostics.Embedded.1:LC.Embedded.
1</VALUE></PROPERTY><PROPERTY NAME="ComponentID"
TYPE="string"><VALUE>25806</VALUE></PROPERTY><PROPERTY NAME="ComponentType"
TYPE="string"><VALUE>APAC</VALUE></PROPERTY><PROPERTY.ARRAY
NAME="ComponentInfoValue"
TYPE="string"><VALUE.ARRAY/></PROPERTY.ARRAY><PROPERTY.ARRAY
NAME="ComponentInfoName"
```

```
TYPE="string"><VALUE.ARRAY/></PROPERTY.ARRAY><PROPERTY.ARRAY
NAME="ComponentInfoTarget"
TYPE="string"><VALUE.ARRAY><VALUE>DCIM:INSTALLED#802__Diagnostics.Embedded.1
:LC.Embedded.1</VALUE></VALUE.ARRAY></PROPERTY.ARRAY><PROPERTY.ARRAY
NAME="ComponentInstalledVersion"
TYPE="string"><VALUE.ARRAY><VALUE>4301A18</VALUE></VALUE.ARRAY></PROPERTY.AR
RAY>

  </INSTANCENAME>
      </VALUE.NAMEDINSTANCE>
      <VALUE.NAMEDINSTANCE>
        <INSTANCENAME CLASSNAME="DCIM_RepoUpdateSWID"><PROPERTY
NAME="Criticality" TYPE="string"><VALUE>1</VALUE></PROPERTY><PROPERTY
NAME="DisplayName" TYPE="string"><VALUE>Dell PERC H330 Mini/Adapter RAID
Controllers firmware version 25.5.5.0005, A11</VALUE></PROPERTY><PROPERTY
NAME="BaseLocation" TYPE="string"><VALUE/></PROPERTY><PROPERTY
NAME="PackagePath" TYPE="string"><VALUE>SAS-
RAID_Firmware_76W42_WN64_25.5.5.0005_A11_01.EXE</VALUE></PROPERTY><PROPERTY
NAME="PackageName" TYPE="string"><VALUE>SAS-
RAID_Firmware_76W42_WN64_25.5.5.0005_A11_01.EXE</VALUE></PROPERTY><PROPERTY
NAME="PackageVersion"
TYPE="string"><VALUE>25.5.5.0005</VALUE></PROPERTY><PROPERTY
NAME="RebootType" TYPE="string"><VALUE>HOST</VALUE></PROPERTY><PROPERTY
NAME="JobID" TYPE="string"><VALUE/></PROPERTY>
    <PROPERTY NAME="Target"
TYPE="string"><VALUE>DCIM:INSTALLED#301_C_RAID.Integrated.1-
1</VALUE></PROPERTY><PROPERTY NAME="ComponentID"
TYPE="string"><VALUE>101551</VALUE></PROPERTY><PROPERTY NAME="ComponentType"
TYPE="string"><VALUE>FRMW</VALUE></PROPERTY><PROPERTY.ARRAY
NAME="ComponentInfoValue"
TYPE="string"><VALUE.ARRAY><VALUE>1000:005F:1028:1F4B</VALUE></VALUE.ARRAY><
/PROPERTY.ARRAY><PROPERTY.ARRAY NAME="ComponentInfoName"
TYPE="string"><VALUE.ARRAY><VALUE>VendorID:DeviceID:SubVendorID:SubDeviceID<
/VALUE></VALUE.ARRAY></PROPERTY.ARRAY><PROPERTY.ARRAY
NAME="ComponentInfoTarget"
TYPE="string"><VALUE.ARRAY><VALUE>DCIM:INSTALLED#301_C_RAID.Integrated.1-
1</VALUE></VALUE.ARRAY></PROPERTY.ARRAY><PROPERTY.ARRAY
NAME="ComponentInstalledVersion"
TYPE="string"><VALUE.ARRAY><VALUE>25.5.4.0006</VALUE></VALUE.ARRAY></PROPERT
Y.ARRAY>

  </INSTANCENAME>
      </VALUE.NAMEDINSTANCE>
      <VALUE.NAMEDINSTANCE>
        <INSTANCENAME CLASSNAME="DCIM_RepoUpdateSWID"><PROPERTY
NAME="Criticality" TYPE="string"><VALUE>1</VALUE></PROPERTY><PROPERTY
NAME="DisplayName" TYPE="string"><VALUE>Dell EMC Server PowerEdge BIOS
R740/R740xd/R640/R940/Precision 7920 Rack Workstation Version
1.6.13</VALUE></PROPERTY><PROPERTY NAME="BaseLocation"
TYPE="string"><VALUE/></PROPERTY><PROPERTY NAME="PackagePath"
TYPE="string"><VALUE>BIOS_YJXXX_WN64_1.6.13.EXE</VALUE></PROPERTY><PROPERTY
```

```
NAME="PackageName"
TYPE="string"><VALUE>BIOS_YJXXX_WN64_1.6.13.EXE</VALUE></PROPERTY><PROPERTY
NAME="PackageVersion"
TYPE="string"><VALUE>1.6.13</VALUE></PROPERTY><PROPERTY NAME="RebootType"
TYPE="string"><VALUE>HOST</VALUE></PROPERTY><PROPERTY NAME="JobID"
TYPE="string"><VALUE/></PROPERTY>
    <PROPERTY NAME="Target"
TYPE="string"><VALUE>DCIM:INSTALLED#741__BIOS.Setup.1-
1</VALUE></PROPERTY><PROPERTY NAME="ComponentID"
TYPE="string"><VALUE>159</VALUE></PROPERTY><PROPERTY NAME="ComponentType"
TYPE="string"><VALUE>BIOS</VALUE></PROPERTY><PROPERTY.ARRAY
NAME="ComponentInfoValue"
TYPE="string"><VALUE.ARRAY/></PROPERTY.ARRAY><PROPERTY.ARRAY
NAME="ComponentInfoName"
TYPE="string"><VALUE.ARRAY/></PROPERTY.ARRAY><PROPERTY.ARRAY
NAME="ComponentInfoTarget"
TYPE="string"><VALUE.ARRAY><VALUE>DCIM:INSTALLED#741__BIOS.Setup.1-
1</VALUE></VALUE.ARRAY></PROPERTY.ARRAY><PROPERTY.ARRAY
NAME="ComponentInstalledVersion"
TYPE="string"><VALUE.ARRAY><VALUE>2.2.0</VALUE></VALUE.ARRAY></PROPERTY.ARRA
Y>

  </INSTANCENAME>
     </VALUE.NAMEDINSTANCE>
    </SIMPLEREQ>
  </MESSAGE>
</CIM>


- WARNING, get repo-based update list data is also copied to file
"repo_based_update_list.xml"
```

3. After the packages in the repository are validated, apply the updates. Notice in the below script output that an overall repository update job will be created followed by an update job for each device that will be updated. After the download of the packages has completed and iDRAC creates the update jobs for all devices detected in the repository, the overall repository job will be marked as completed. Then each update job will execute; devices such as DIAGS and Lifecycle Controller driver packs that support immediate updates will run first and devices such as BIOS and PERC that require a server reboot to apply the update are run later.

```
C:\Python27>InstallFromRepositoryREDFISH.py -ip 192.168.0.120 -u root -p
calvin -i y --ipaddress 192.168.0.130 --sharetype NFS --sharename
/nfs/R640_repo_updates --applyupdate True --rebootneeded True
- WARNING, arguments and values for InstallFromRepository method
ShareType: NFS
RebootNeeded: True
ApplyUpdate: True
ShareName: /nfs/R640_repo_updates
IPAddress: 100.65.84.150
```

```
- PASS: POST command passed for method "InstallFromRepository", status code
202 returned
- PASS, job ID JID_513763153006 successfully created
- WARNING, Job ID JID_513763153006 not marked completed, current status:
"Downloading the Catalog.xml update package.", job polling time: "0:00:00"
- WARNING, Job ID JID_513763153006 not marked completed, current status:
"Package successfully downloaded.", job polling time: "0:00:06"
- WARNING, Job ID JID_513763153006 not marked completed, current status:
"Package successfully downloaded.", job polling time: "0:00:11"
- WARNING, Job ID JID_513763153006 not marked completed, current status:
"Package successfully downloaded.", job polling time: "0:00:17"
- WARNING, Job ID JID_513763153006 not marked completed, current status:
"Package successfully downloaded.", job polling time: "0:00:22"

- PASS, job ID JID_513763153006 successfully marked completed

- Final detailed job results -

@odata.type: #DellJob.v1_0_1.DellJob
JobState: Completed
Description: Job Instance
TargetSettingsURI: None
@odata.id: /redfish/v1/Managers/iDRAC.Embedded.1/Jobs/JID_513763153006
@odata.context: /redfish/v1/$metadata#DellJob.DellJob
MessageArgs: [u'NA']
CompletionTime: 2019-02-28T11:53:01
PercentComplete: 100
StartTime: TIME_NOW
MessageId: RED001
Message: Job completed successfully.
EndTime: None
Id: JID_513763153006
JobType: RepositoryUpdate
Name: Repository Update

- WARNING, repository update job marked completed. Checking now to see if
any update jobs were created due to different firmware versions detected

- WARNING, scheduled update job ID detected, server rebooting to apply the
update(s)
- WARNING, Job ID JID_513763218603 not marked completed, current status:
"Task successfully scheduled.", job polling time: "0:00:00"
...
- WARNING, Job ID JID_513763218603 not marked completed, current status:
"The specified job is in progress.", job polling time: "0:09:19"

- PASS, job ID JID_513763218603 successfully marked completed

- Final detailed job results -

@odata.type: #DellJob.v1_0_1.DellJob
```

```
JobState: Completed
Description: Job Instance
TargetSettingsURI: None
@odata.id: /redfish/v1/Managers/iDRAC.Embedded.1/Jobs/JID_513763218603
@odata.context: /redfish/v1/$metadata#DellJob.DellJob
MessageArgs: []
CompletionTime: 2019-02-28T12:02:39
PercentComplete: 100
StartTime: TIME_NOW
MessageId: PR19
Message: The specified job has completed successfully.
EndTime: TIME_NA
Id: JID_513763218603
JobType: FirmwareUpdate
Name: update:DCIM:INSTALLED#741__BIOS.Setup.1-1

- PASS, job ID JID_513763429621 successfully marked completed

- Final detailed job results -

@odata.type: #DellJob.v1_0_1.DellJob
JobState: Completed
Description: Job Instance
TargetSettingsURI: None
@odata.id: /redfish/v1/Managers/iDRAC.Embedded.1/Jobs/JID_513763429621
@odata.context: /redfish/v1/$metadata#DellJob.DellJob
MessageArgs: []
CompletionTime: 2019-02-28T12:02:38
PercentComplete: 100
StartTime: TIME_NOW
MessageId: PR19
Message: Job completed successfully.
EndTime: TIME_NA
Id: JID_513763429621
JobType: FirmwareUpdate
Name: update:DCIM:INSTALLED#301_C_RAID.Integrated.1-1

- PASS, job ID JID_513763590675 successfully marked completed

- Final detailed job results -

@odata.type: #DellJob.v1_0_1.DellJob
JobState: Completed
Description: Job Instance
TargetSettingsURI: None
@odata.id: /redfish/v1/Managers/iDRAC.Embedded.1/Jobs/JID_513763590675
@odata.context: /redfish/v1/$metadata#DellJob.DellJob
MessageArgs: []
CompletionTime: 2019-02-28T11:52:52
PercentComplete: 100
```

```
StartTime: TIME_NOW
MessageId: RED001
Message: Job completed successfully.
EndTime: None
Id: JID_513763590675
JobType: FirmwareUpdate
Name: update:DCIM:INSTALLED#802__Diagnostics.Embedded.1:LC.Embedded.1
```

4. Check the firmware versions of the devices that have been updated. In the output below, you will notice DIAGs, BIOS, and PERC are updated to new versions.

```
C:\Python27>InstallFromRepositoryREDFISH.py -ip 192.168.0.120 -u root -p
calvin -g y
- WARNING, current devices detected with firmware version and updateable
status -
Device Name: PCIe SSD in Slot 2 in Bay 1, Firmware Version: 1.0.0,
Updatable: True
Device Name: BOSS-S1, Firmware Version: 2.5.13.3016, Updatable: True
Device Name: PERC H330 Mini, Firmware Version: 25.5.5.0005, Updatable: True
Device Name: OS Collector, Firmware Version: 0, Updatable: True
Device Name: PCIe SSD in Slot 2, Firmware Version: KPYABD3Q, Updatable: True
Device Name: BP14G+EXP 0:1, Firmware Version: 2.40, Updatable: True
Device Name: iDRAC Service Module Installer, Firmware Version: 0, Updatable:
True
Device Name: Power Supply.Slot.1, Firmware Version: 00.23.32, Updatable:
True
Device Name: Disk 0 on AHCI Controller in slot 1, Firmware Version: DL43,
Updatable: True
Device Name: BIOS, Firmware Version: 1.6.13, Updatable: True
Device Name: OS Drivers Pack, Firmware Version: 0, Updatable: True
Device Name: Disk 1 in Backplane 1 of Integrated RAID Controller 1, Firmware
Version: K774, Updatable: True
Device Name: Integrated Dell Remote Access Controller, Firmware Version:
3.30.30.30, Updatable: True
Device Name: Dell 64 Bit uEFI Diagnostics, version 4301, 4301A25, 4301.26,
Firmware Version: 4301A25, Updatable: True
Device Name: Disk 0 in Backplane 1 of Integrated RAID Controller 1, Firmware
Version: FSF9, Updatable: True
Device Name: QLogic 577xx/578xx 10 Gb Ethernet BCM57800 - 18:66:DA:8E:28:24,
Firmware Version: 08.07.00, Updatable: True
Device Name: System CPLD, Firmware Version: 1.0.2, Updatable: True
Device Name: Lifecycle Controller, Firmware Version: 3.30.30.30, Updatable:
False
```

3.7.4 Update the repository firmware by using Dell EMC Repository

Using the InstallFromRepository Dell EMC OEM method, you can also point to the Dell EMC repository which simplfies the process of updating your servers to the latest firmware versions without the need of creating a custom repository.

The example below points to the Dell EMC repository http://downloads.dell.com or IP `143.166.147.76`. If required, you can reference the complete workflow of using the `InstallFromRepository` script with a custom reository above—the workflows are identicial, but uses a different repository.

```
C:\Python27>InstallFromRepositoryREDFISH.py -ip 192.168.0.120 -u root -p
calvin -i y --ipaddress 143.166.147.76 --sharetype HTTP --applyupdate False
- WARNING, arguments and values for InstallFromRepository method
ShareType: HTTP
IPAddress: 143.166.147.76
ApplyUpdate: False
- PASS: POST command passed for method "InstallFromRepository", status code
202 returned
- PASS, job ID JID_513782140871 successfully created
- WARNING, Job ID JID_513782140871 not marked completed, current status:
"Downloading the Catalog.xml update package.", job polling time: "0:00:00"
- PASS, job ID JID_513782140871 successfully marked completed
- Final detailed job results -
@odata.type: #DellJob.v1_0_1.DellJob
JobState: Completed
Description: Job Instance
TargetSettingsURI: None
@odata.id: /redfish/v1/Managers/iDRAC.Embedded.1/Jobs/JID_513782140871
@odata.context: /redfish/v1/$metadata#DellJob.DellJob
MessageArgs: [u'NA']
CompletionTime: 2019-02-28T12:23:43
PercentComplete: 100
StartTime: TIME_NOW
MessageId: RED001
Message: Job completed successfully.
EndTime: None
Id: JID_513782140871
JobType: RepositoryUpdate
Name: Repository Update

- WARNING, "ApplyUpdate = False" selected, execute script with -r agrument
to view the repo update list which will report devices detected for firmware
updates
```

3.7.5 InstallFromURI

The InstallFromURI method is used for creating a job for the update service. The successful execution of this method creates a job.

InstallFromURI Method Parameters

Qualifiers	Name	Type	Description
IN	IgnoreCertWarning	Edm.String	Specifies if certificate warning should be ignored when HTTPS is used. If IgnoreCertWarning is On, warnings are ignored. Default is 2 (On).
IN	ProxyPasswd	Edm.String	The password for the proxy server.
IN	ProxyPort	Edm.String	Port for the proxy server. Default is set to 80.
IN	ProxyServer	Edm.String	The IP address of the proxy server.
IN	ProxySupport	Edm.String	Specifies if a proxy should be used. Default is 1 (Off).
IN	ProxyType	Edm.String	The type of the proxy server. Default is 0 (HTTP).
IN	ProxyUname	Edm.String	The user name for the proxy server.
IN, Required	Target	Edm.String	A Link to the DellSoftwareInventory resource whose firmware needs to be updated. It shall be the @odata.id of the DellSoftwareInventory Resource.
IN, Required	URI	Edm.String	Network file location of the firmware to be installed. Supported network share types are HTTP, HTTPS, CIFS, NFS, TFTP, and FTP. Special characters must be encoded in the URI format.
OUT	Job	Edm.String	Reference to the job spawned if the operation continues after the method returns.
OUT	Message	Edm.String	Error or information message corresponding to the MessageID is returned, in English.
OUT	MessageArguments	Edm.String	Substitution variables for dynamic error or information messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.8 DelliDRACCardService

The DelliDRACCardService resource provides actions to support iDRAC configuration.

URI: `/redfish/v1/Dell/Managers/iDRAC.Embedded.1/DelliDRACCardService`

3.8.1 ExportSSLCertificate

This method is used to export the SSL certificate from the iDRAC, based on input parameter type. ExportSSLCertificate returns the certificate.

ExportSSLCertificate Method Parameters

Qualifiers	Name	Type	Description
IN, Required	SSLCertType	Edm.String	The type of the certificate to be exported.
OUT	CertificateFile	Edm.String	Exported certificate file encoded in base64.
OUT	Message	Edm.String	Error or information message in English corresponding to the message ID is returned.
OUT	MessageArgs	Edm.String	Substitution variables for dynamic error messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.8.2 ImportSSLCertificate

This method is used to import the SSL certificate to iDRAC, based on the input parameter Ttype. After importing the certificate, the iDRAC will automatically restart.

ImportSSLCertificate Method Parameters

Qualifiers	Name	Type	Description
IN, Required	CertificateType	Edm.String	Type of the certificate to be imported.
IN	Passphrase	Edm.String	A passphrase for certificate file. Note: This is optional parameter for CSC certificate, and not required for Server and CA certificates.
IN, Required	SSLCertificateFile	Edm.String	A base-64 encoded string of the XML Certificate file. Note: For importing CSC certificate, user has to convert PKCS file to base64 format. Use the openssl command. The CTC file content must be in PEM the format (base-64 encoded).
OUT	Message	Edm.String	Error information message in English corresponding to the message ID is returned.
OUT	MessageArgs	Edm.String	Substitution variables for dynamic error messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.8.3 iDRACReset

This method is used to reset iDRAC.

iDRACReset Method Parameters

Qualifiers	Name	Type	Description
IN	Force	Edm.String	This option is used to reset the iDRAC by force or gracefully.
OUT	Message	Edm.String	Error information message in English corresponding to the message ID is returned.
OUT	MessageArgs	Edm.String	Substitution variables for dynamic error messages.
OUT	MessageID	Edm.String	The message ID for the output message.

3.8.4 iDRACResetCfg

This method is used to reset the iDRAC to factory default configurations.

iDRACResetCfg Method Parameters

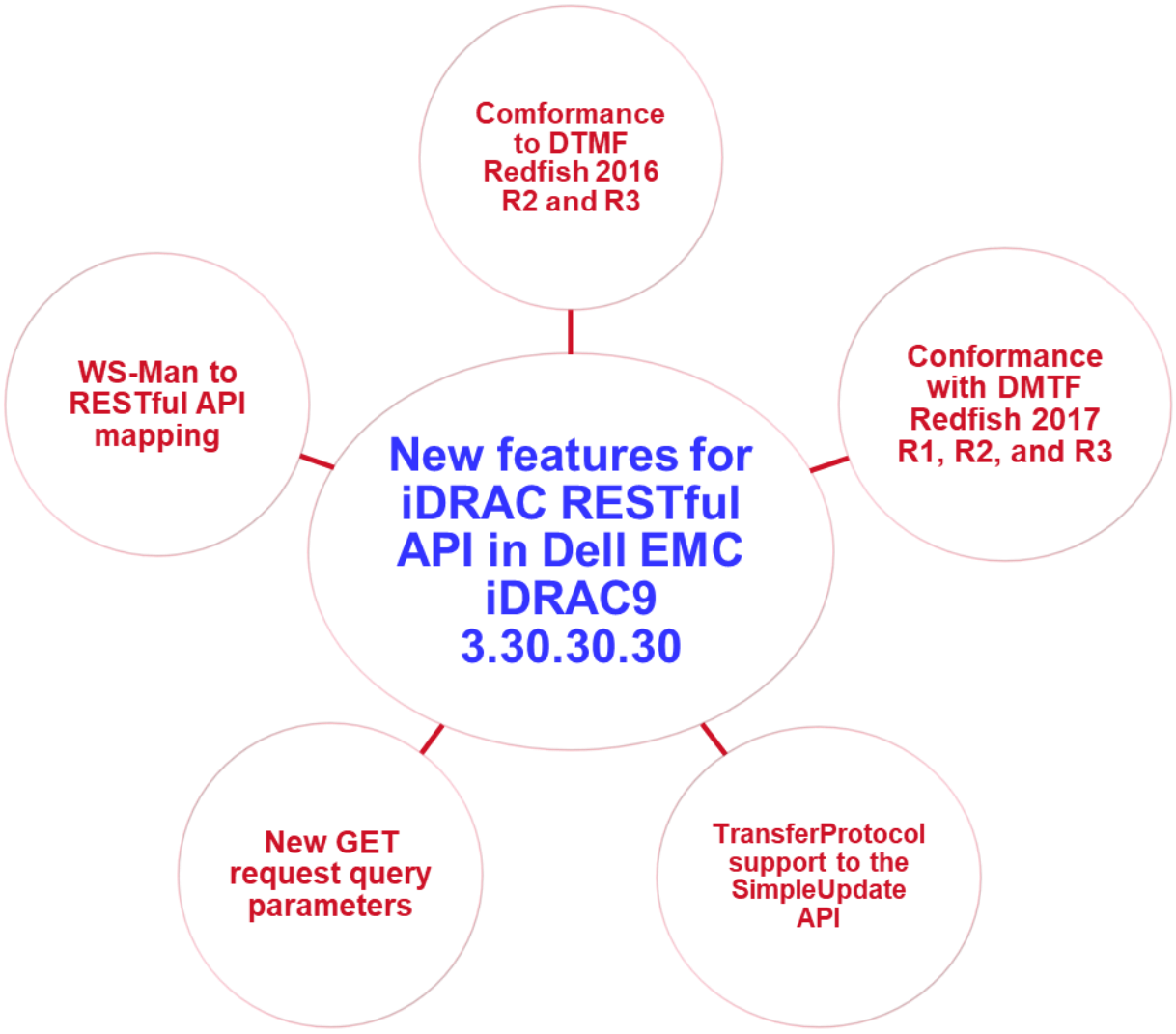
Qualifiers	Name	Type	Description
IN	Force	Edm.String	This option is used to reset the iDRAC to factory defaults by force or gracefully.
IN	Preserve	Edm.String	• Preserve = 0 (Default)—Reset all configuration to default except network and users. • Preserve = 1 (All)—Reset all configuration to default including network and users.

			• Preserve = 2 (ResetAllWithRootDefaults) —Reset all configuration to default including network preserve default user as `root/calvin`.
OUT	Message	Edm.String	Error information message in English corresponding to the message ID is returned.
OUT	MessageArgs	Edm.String	Substitution variables for dynamic error messages.
OUT	MessageID	Edm.String	The message ID for the output message.

4 Summary

The DMTF Redfish standard is emerging as a key new tool for efficient, scalable, and secure server management. Utilizing an industry-standard interface and data format, Redfish supports rapid development of automation for one-to-many server management. System administrators and IT developers will appreciate Redfish's features that can increase efficiency, lower costs and boost productivity across their organizations.

Dell EMC is a committed leader in the development and implementation of open, industry standards. Supporting Redfish within the iDRAC with Lifecycle Controller further enhances the manageability of PowerEdge servers, providing another powerful tool to help IT administrators reduce complexity and help save time and money.



A Map DRAC RESTful API to WS-Man—Phase I

The following table details the mapping of iDRAC-supported WS-Man classes to iDRAC RESTful API resources and URIs delivered in iDRAC9 firmware 3.30.30.30.

WS-Man Class	iDRAC RESTful Resource	URI
DCIM_ControllerView	DellController	"/redfish/v1/Dell/Systems/Sysem.Emdedded.1/Storage/DellController(SelectorSet)"
DCIM_ControllerView	DellControllerCollection	"/redfish/v1/Dell/Systems/Sysem.Emdedded.1/Storage/DellControllerCollection"
DCIM_CPUView	DellProctexessor	"/redfish/v1/Dell/Systems/System.Emdedded.1/Processors/DellProcessor(SelectorSet)"
DCIM_CPUView	DellProcessorCollection	"/redfish/v1/Dell/Systems/System.Emdedded.1/Processors/DellProcessorCollection"
DCIM_EnclosureEMMView	DellEnclosureEMM	"/redfish/v1/Dell/Chassis/System.Embedded.1/DellEnclosureEMM(SelectorSet)"
DCIM_EnclosureEMMView	DellEnclosureEMMCollection	"/redfish/v1/Dell/Chassis/System.Embedded.1/DellEnclosureEMMCollection"
DCIM_EnclosureFanSensor	DellEnclosureFanSensor	"/redfish/v1/Dell/Chassis/System.Emdedded.1/DellEnclosureFanSensor(SelectionSet)"
DCIM_EnclosureFanSensor	DellEnclosureFanSensorCollection	"/redfish/v1/Dell/Chassis/System.Embedded.1/DellEnclosureFanSensorCollection"
DCIM_EnclosurePSUView	DellEnclosurePowerSupply	"/redfish/v1/Dell/Chassis/System.Embedded.1/DellEnclosurePowerSupply(SelectorSet)"
DCIM_EnclosurePSUView	DellEnclosurePowerSupplyCollection	"/redfish/v1/Dell/Chassis/System.Embedded.1/DellEnclosurePowerSupplyCollection"
DCIM_EnclosureTemperatureSensor	DellEnclosureTemperatureSensor	"/redfish/v1/Dell/Chassis/System.Embedded.1/DellEnclosureTemperatureSensor"
DCIM_EnclosureTemperatureSensor	DellEnclosureTemperatureSensorCollection	"/redfish/v1/Dell/Chassis/System.Embedded.1/DellEnclosureTemperatureSensorCollection"
DCIM_EnclosureView	DellEnclosure	"/redfish/v1/Dell/Chassis/System.Embedded.1/DellEnclosure(SelectorSet)"
DCIM_EnclosureView	DellEnclosureCollection	"/redfish/v1/Dell/Chassis/System.Embedded.1/DellEnclosureCollection"
DCIM_FCStatistics	DellFCStatistics	"/redfish/v1/Dell/Chassis/System.Embedded.1/NetworkPorts/DellFCStatistics(SelectorSet)"
DCIM_FCStatistics	DellFCStatisticsCollection	"/redfish/v1/Dell/Chassis/System.Embedded.1/NetworkPorts/DellFCStatisticsCollection"
DCIM_FCView	DellFC	"/redfish/v1/Dell/Chassis/System.Embedded.1/NetworkAdapters/DellFC(SelectorSet)"
DCIM_FCView	DellFCCollection	"/redfish/v1/Dell/Chassis/System.Embedded.1/NetworkAdapters/DellFCCollection"
DCIM_iDRACCardService	DelliDRACCardService	"/redfish/v1/Dell/Managers/iDRAC.Embedded.1/DelliDRACCardService"
DCIM_iDRACCardView	DelliDRACCard	"/redfish/v1/Dell/Managers/iDRAC.Embedded.1/DelliDRACCard(SelectorSet)"
DCIM_iDRACCardView	DelliDRACCardCollection	"/redfish/v1/Dell/Managers/iDRAC.Embedded.1/DelliDRACCardCollection"

WS-Man Class	iDRAC RESTful Resource	URI
DCIM_JobService	DellJobService	"/redfish/v1/Dell/Managers/iDRAC.Embedded.1/DellJobService"
DCIM_LCService	DellLCService	"/redfish/v1/Dell/Managers/iDRAC.Embedded.1/DellLCService"
DCIM_License	DellLicense	"/redfish/v1/Dell/Managers/iDRAC.Embedded.1/DellLicense(SelectorSet)"
DCIM_License	DellLicenseCollection	"/redfish/v1/Dell/Managers/iDRAC.Embedded.1/DellLicenseCollection"
DCIM_LicenseManagementService	DellLicenseManagementService	"/redfish/v1/Dell/Managers/iDRAC.Embedded.1/DellLicenseManagementService"
DCIM_MemoryView	DellMemory	"/redfish/v1/Dell/Systems/System.Embedded.1/Memory/DellMemory(SelectorSet)"
DCIM_MemoryView	DellMemoryCollection	"/redfish/v1/Dell/Systems/System.Embedded.1/Memory/DellMemoryCollection"
DCIM_NICCapabilities	DellNICCapabilities	"/redfish/v1/Dell/Systems/System.Embedded.1/NetworkDeviceFunctions/DellNICCapabilities(SelectorSet)"
DCIM_NICCapabilities	DellNICCapabilitiesCollection	"/redfish/v1/Dell/Systems/System.Embedded.1/NetworkDeviceFunctions/DellNICCapabilitiesCollection"
DCIM_NICStatistics	DellNICStatistics	"/redfish/v1/Dell/Systems/System.Embedded.1/NetworkAdapters/NetworkDeviceFunctions/DellNICStatistics(SelectorSet)"
DCIM_NICStatistics	DellNICStatisticsCollection	"/redfish/v1/Dell/Systems/System.Embedded.1/NetworkAdapters/NetworkDeviceFunctions/DellNICStatisticsCollection"
DCIM_NICView	DellNIC	"/redfish/v1/Dell/Systems/System.Embedded.1/NetworkDeviceFunctions/DellNIC(SelectorSet)"
DCIM_NICView	DellNICCollection	"/redfish/v1/Dell/Systems/System.Embedded.1/NetworkDeviceFunctions/DellNICCollection"
DCIM_NumericSensor	DellNumericSensor	"/redfish/v1/Dell/Systems/System.Embedded.1/DellNumericSensor(SelectorSet)"
DCIM_NumericSensor	DellNumericSensorCollection	"/redfish/v1/Dell/Systems/System.Embedded.1/DellNumericSensorCollection"
DCIM_OSDeploymentService	DellOSDeploymentService	"/redfish/v1/Dell/Systems/System.Embedded.1/DellOSDeploymentService"
DCIM_PCIDeviceView	DellPCIDevice	"/redfish/v1/Dell/Systems/System.Emdedded.1/DellPCIDevice(SelectorSet)"
DCIM_PCIDeviceView	DellPCIDeviceCollection	"/redfish/v1/Dell/Systems/System.Embedded.1/DellPCIDeviceCollection"
DCIM_PCIeSSDBackPlaneView	DellPCIeSSDBackPlane	"/redfish/v1/Dell/Systems/System.Embedded.1/Storage/DellPCIeSSDBackPlane(SelectorSet)"
DCIM_PCIeSSDBackPlaneView	DellPCIeSSDBackPlaneCollection	"/redfish/v1/Dell/Systems/System.Embedded.1/Storage/DellPCIeSSDBackPlaneCollection"
DCIM_PCIeSSDExtenderView	DellPCIeSSDExtender	"/redfish/v1/Dell/Systems/System.Embedded.1/Storage/DellPCIeSSDExtender(SelectorSet)"
DCIM_PCIeSSDExtenderView	DellPCIeSSDExtenderCollection	"/redfish/v1/Dell/Systems/System.Embedded.1/Storage/DellPCIeSSDExtenderCollection"
DCIM_PCIeSSDView	DellPCIeSSD	"/redfish/v1/Dell/Systems/System.Embedded.1/Storage/DellPCIeSSD(SelectorSet)"

WS-Man Class	iDRAC RESTful Resource	URI
DCIM_PCIeSSDView	DellPCIeSSDCollection	"/redfish/v1/Dell/Systems/System.Embedded.1/Storage/DellPCIeSSDCollection"
DCIM_PersistentStorageService	DellPersistentStorageService	"/redfish/v1/Dell/Mangers/iDRAC.Embedded.1/DellPersistentStorageService(SelectorSet)"
DCIM_PhysicalDiskView	DellPhysicalDisk	"/redfish/v1/Dell/Systems/System.Embedded.1/Storage/Drives/DellPhysicalDisk(SelectorSet)"
DCIM_PhysicalDiskView	DellPhysicalDiskCollection	"/redfish/v1/Dell/Systems/System.Embedded.1/Storage/Drives/DellPhysicalDiskCollection"
DCIM_PowerSupply	DellPowerSupply	"/redfish/v1/Dell/Chassis/System.Embedded.1/Power/PowerSupplies/DellPowerSupply(SelectorSet)"
DCIM_PowerSupply	DellPowerSupplyCollection	"/redfish/v1/Dell/Chassis/System.Embedded.1/Power/PowerSupplies/DellPowerSupplyCollection"
DCIM_PowerSupplyView	DellPowerSupplyView	"/redfish/v1/Dell/Chassis/System.Embedded.1/Power/PowerSupplies/DellPowerSupplyView(SelectorSet)"
DCIM_PowerSupplyView	DellPowerSupplyViewCollection	"/redfish/v1/Dell/Chassis/System.Embedded.1/Power/PowerSupplies/DellPowerSupplyViewCollection"
DCIM_PresenceAndStatusSensor	DellPresenceAndStatusSensor	"/redfish/v1/Dell/Systems/System.Embedded.1/DellPresenceAndStatusSensor(SelectorSet)"
DCIM_PresenceAndStatusSensor	DellPresenceAndStatusSensorCollection	"/redfish/v1/Dell/Systems/System.Embedded.1/DellPresenceAndStatusSensorCollection"
DCIM_PSNumericSensor	DellPSNumericSensor	"/redfish/v1/Dell/Systems/System.Embedded.1/DellPSNumericSensor(SelectorSet)"
DCIM_PSNumericSensor	DellPSNumericSensorCollection	"/redfish/v1/Dell/Systems/System.Embedded.1/DellPSNumericSensorCollection"
DCIM_RAIDService	DellRaidService	"/redfish/v1/Dell/Systems/System.Embedded.1/DellRaidService(SelectorSet)"
DCIM_Sensor	DellSensor	"/redfish/v1/Dell/Systems/System.Embedded.1/DellSensor(Selector Set)"
DCIM_Sensor	DellSensorCollection	"/redfish/v1/Dell/Systems/System.Embedded.1/DellSensorCollection"
DCIM_SoftwareIdentity	DellSoftwareInventory	"/redfish/v1/Dell/UpdateService/FirmwareInventory/DellSoftwareInventory(SelectorSet)"
DCIM_SoftwareIdentity	DellSoftwareInventoryCollection	"/redfish/v1/Dell/UpdateService/FirmwareInventory/DellSoftwareInventoryCollection"
DCIM_SoftwareInstallationService	DellSoftwareInstallationService	"/redfish/v1/Dell/Systems/System.Embedded.1/DellSoftwareInstallationService"
DCIM_SwitchConnectionView	DellSwitchConnection	"/redfish/v1/Dell/Systems/System.Embedded.1/NetworkPorts/DellSwitchConnection(SelectorSet)"
DCIM_SwitchConnectionView	DellSwitchConnectionCollection	"/redfish/v1/Dell/Systems/System.Embedded.1/NetworkPorts/DellSwitchConnectionCollection"
DCIM_SystemView	DellSystem	"/redfish/v1/Dell/Systems/System.Embedded.1/DellSystem(SelectorSet)"
DCIM_SystemView	DellSystemCollection	"/redfish/v1/Dell/Systems/System.Embedded.1/DellSystemCollection"
DCIM_VFlashView	DellvFlash	"/redfish/v1/Dell/Managers/iDRAC.Embedded.1/DellvFlash(SelectorSet)"

WS-Man Class	iDRAC RESTful Resource	URI
DCIM_VFlashView	DellvFlashCollection	"/redfish/v1/Dell/Managers/iDRAC.Embedded.1/DellvFlashCollection"
DCIM_VirtualDiskView	DellVirtualDisk	"/redfish/v1/Dell/Systems/System.Embedded.1/Storage/Volumes/DellVirtualDisk(SelecorSet)"
DCIM_VirtualDiskView	DellVirtualDiskCollection	"/redfish/v1/Dell/Systems/System.Embedded.1/Storage/Volumes/DellVirtualDiskCollection"

B Additional Resources

Below are links to Dell EMC iDRAC RESTful API white papers, DMTF white papers, specifications, and other useful documents

- Introducing the Dell EMC PowerEdge Redfish API white paper
  http://en.community.dell.com/techcenter/extras/m/white_papers/20442330
- RESTful Server Configuration with iDRAC RESTful API white paper
  http://en.community.dell.com/techcenter/extras/m/white_papers/20443207
- Standards-based storage management with iDRAC RESTful API:
  https://downloads.dell.com/manuals/all-products/esuprt_solutions_int/esuprt_solutions_int_solutions_resources/dell-management-solution-resources_white-papers10_en-us.pdf
- Standards-based server networking management with iDRAC RESTful API:
  https://downloads.dell.com/manuals/all-products/esuprt_software/esuprt_it_ops_datcentr_mgmt/dell-management-solution-resources_white-papers8_en-us.pdf
- Automating Redfish with PowerShell white paper:
  http://en.community.dell.com/techcenter/extras/m/white_papers/20444434
- Dell EMC Open Source Redfish PowerShell and Python Scripting
  http://github.com/dell/iDRAC-Redfish-Scripting
- OpenManage Ansible Modules
  https://github.com/dell/Dell-EMC-Ansible-Modules-for-iDRAC
- Dell EMC Techcenter YouTube channel for iDRAC RESTful API
  https://www.youtube.com/watch?v=d90mX1W51S0&list=PLe5xhhyFjDPcUPEcwVd--1pS17nFNiHTd

- Introduction to Redfish presentation:
  http://www.dmtf.org/sites/default/files/SPMF%20Introduction%20to%20Redfish%20May%202016.pdf
- DMTF Redfish specification
  http://www.dmtf.org/sites/default/files/standards/documents/DSP0266_1.0.1.pdf
- Webinar – Redfish overview: https://www.brighttalk.com/webcast/9077/156709
- YouTube Redfish School: https://www.youtube.com/dmtforg