DELL EMC

White paper

Using Server Configuration Profiles to Deploy Operating Systems to Dell EMC PowerEdge Servers

Abstract

This white paper describes best practices to install an operating system automatically via a Server Configuration Profile. This method can be deployed to multiple servers, greatly reducing the time required to get a server up and running.

April 2020

Revisions

Date	Description
April 2020	Initial release

Acknowledgements

This paper was produced by the following:

Authors: Dallin Strong, Charles Stracener, Texas Roemer

Support: Sanjay Rao, Steve Fagan, Krishna Kanukurthy, and Chandran Venkatachal

The information in this publication is provided “as is.” Dell Inc. makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.

Table of contents

Executive Summary

One of the most time-consuming steps in getting a server ready for use is installing the operating system. This document describes how to use the already existing Server Configuration Profile (SCP) to deploy the operating system from an ISO file to a local PowerEdge server. When both SCP profile and ISO file are stored on a network share, they can be leveraged in a simple, efficient process.

For more details on SCP functionality, see Server cloning by using Server Configuration Profiles (SCP) on Dell EMC PowerEdge servers.

1 Server configuration profile template operating system deployment additions

For this feature, the following attributes have been added to the SCP template and can be seen in the resulting XML/JSON file when doing an SCP export operation. These attributes can be found under the LifecycleController component.

NOTE: These attributes act as a template for executing the OSD operation and will not contain values. These attributes do not store any values during an SCP import operation.

Table 1 lists each new field and an accompanying definition.

Table 3 Added fields

Attribute	Definition
SupportedOSList	A read-only field which shows up as a commented line and contains a list of operating systems for which the system has drivers available. If this field is blank the operating system driver pack will need to be installed. To do this go to www.dell.com/support and search for your server model, and then go to **Drivers & Downloads** and look for the **Dell OS Driver Pack**. This is an executable file that contains the drivers for operating system installation, and it must be installed on the server.
OSName	The name of the specific operating system to install, this attribute is used to select the operating system drivers that are used for operating system installation. The operating system name must be one contained in the SupportedOSList attribute.
OSMediaShareIP	The IP address of the share that contains the operating system installation file.
OSMediaShareName	The name of the share that contains the operating system installation file.
OSMediaShareUsername	The username of the share that contains the operating system installation file. **NOTE:** Only CIFS shares will use the media share username and password. For others this field will remain blank.
OSMediaSharePassword	The password of the share that contains the operating system installation file. **NOTE**: Only CIFS shares will use the media share username and password. For others this field will remain blank.

OSMediaShareDomainName	Domain name for the network on which the installation file is located. Can be left blank if network has no domain name.
OSMediaShareType	Share type of the share on which the operating system installation file is located. Supported share types include CIFS, NFS, HTTP, HTTPS, or local. When local is used as the share type, the SCP network information is used as the share data. The operating system media must be stored in the same location on the network share as the SCP template. Other operating system media-related attributes are ignored.
OSMediaName	Name of the operating system installation file
AnswerFileName	Answer file name, or if a Linux operating system is being installed this will be the name of the kickstarter file. This file is located in the same directory as the operating system installation file. It contains information regarding the operating system install such as time zone and user account setup that will be applied automatically during the install process. It does not need to be entered manually.
ExposeDuration	Amount of time (in seconds) that the driver partition is exposed. Default is 65535 seconds, or 18 hours.
OSMediaHashType	If the operating system has a hash value to verify against, this field contains the hash type. Supported hash types are MD5 and SHA1.
OSMediaHashValue	Hash value used to verify the operating system installation media.

XML snippet example:

```
  <Component FQDD="LifecycleController.Embedded.1">
      <!-- <Attribute Name="OSD.1#SupportedOSList">Microsoft Windows Server
2016</Attribute> -->
    <Attribute Name="OSD.1#OSName"></Attribute>
    <Attribute Name="OSD.1#OSMediaShareIP"></Attribute>
    <Attribute Name="OSD.1#OSMediaShareName"></Attribute>
    <Attribute Name="OSD.1#OSMediaShareUsername"></Attribute>
    <Attribute Name="OSD.1#OSMediaSharePassword"></Attribute>
    <Attribute Name="OSD.1#OSMediaShareDomainName"></Attribute>
    <Attribute Name="OSD.1#OSMediaShareType"></Attribute>
    <Attribute Name="OSD.1#OSMediaName"></Attribute>
    <Attribute Name="OSD.1#AnswerFileName"></Attribute>
    <Attribute Name="OSD.1#ExposeDuration"></Attribute>
    <Attribute Name="OSD.1#OSMediaHashType"></Attribute>
    <Attribute Name="OSD.1#OSMediaHashValue"></Attribute>
  </Component>
```

JSON snippet example:

```
    { "Name": "OSD.1#SupportedOSList", "Value": “Microsoft Windows Server
2016",
   "Set On Import": "False", "Comment": "Read and Write" },
    { "Name": "OSD.1#OSName", "Value": "Microsoft Windows Server 2016",
      "Set On Import": "True", "Comment": "Read and Write" },
    { "Name": "OSD.1#OSMediaShareIP", "Value": “",
      "Set On Import": "True", "Comment": "Read and Write" },
    { "Name": "OSD.1#OSMediaShareName", "Value": “",
      "Set On Import": "True", "Comment": "Read and Write" },
    { "Name": "OSD.1#OSMediaShareUsername", "Value": “",
      "Set On Import": "True", "Comment": "Read and Write" },
    { "Name": "OSD.1#OSMediaSharePassword", "Value": “",
      "Set On Import": "True", "Comment": "Read and Write" },
    { "Name": "OSD.1#OSMediaShareType", "Value": “",
      "Set On Import": "True", "Comment": "Read and Write" },
    { "Name": "OSD.1#OSMediaName", "Value": “",
      "Set On Import": "True", "Comment": "Read and Write" },
    { "Name": "OSD.1#AnswerFileName", "Value": "",
      "Set On Import": "True", "Comment": "Read and Write" },
    { "Name": "OSD.1#ExposeDuration", "Value": “",
      "Set On Import": "True", "Comment": "Read and Write" },
    { "Name": "OSD.1#OSMediaHashType", "Value": "",
      "Set On Import": "True", "Comment": "Read and Write" },
    { "Name": "OSD.1#OSMediaHashValue", "Value": "",
      "Set On Import": "True", "Comment": "Read and Write" }
```

2 Server configuration profile operations

The following section provides a general overview of the supported SCP operations for all supported iDRAC interfaces. These operations include export, import preview and import.

2.1 Export server configuration profile template

To use the server configuration profile feature, first create or export an SCP file that contains all the necessary server configuration information. The following examples show how to perform an SCP export using all supported iDRAC interfaces.

NOTE: After executing the export operation using any supported iDRAC interface, a job ID will be returned which you can query to check the status and validate if the process completed or failed. If failed, check the iDRAC Lifecycle Logs for more details. For more details on checking the job status, refer to the workflow section in this document.

2.1.1 SCP export using the iDRAC UI:

To export an SCP profile using the iDRAC UI, go to **Configuration>Server Configuration Profile>Export**, fill in the fields as required, then click **Export**.

2.1.2 SCP export using RACADM:

To export an SCP profile using RACADM, use the `racadm get` command. For more examples or help, execute `racadm help get`.

RACADM example:

```
C:\>racadm -r 192.168.0.120 -u root -p calvin get -f R740_scp_file.xml -t xml
-l 192.168.0.130:/nfs
```

2.1.3 SCP export using Redfish:

To export an SCP profile using Redfish, execute a POST call on the OEM action, `EID_674_Manager.ExportSystemConfiguration`.

Redfish POST example:

```
URI:
/redfish/v1/Managers/iDRAC.Embedded.1/Actions/Oem/EID_674_Manager.ExportSyste
mConfiguration
Header: content-type application/json
Body: {'ExportFormat': 'XML', 'ShareParameters': {'ShareType': 'NFS',
'ShareName': '/nfs', 'IPAddress': '192.168.0.130', 'Target': 'ALL',
'FileName': 'R740_scp_file.xml'}}
```

2.2 Import preview server configuration profile template

Before importing the SCP file, it is not required, but is recommended, that you execute the import preview operation. By executing this operation first, you validate your SCP file format, and that the current values of attributes that you changed are passed upon application. If you do not execute an import preview first and you import an SCP file with issues, you must wait until the job is either marked failed or completed with errors and then debug the issue and run the import operation again.

NOTE: After executing an import preview operation using any supported iDRAC interface, a job ID is returned which you can query to check the status and validate if the process completed or failed. If failed, check your iDRAC Lifecycle Logs for more details. For more details on checking the job status, refer to the workflow section in this document.

2.2.1 SCP import preview using iDRAC UI:

To import preview a SCP profile using the iDRAC UI, go to **Configuration>Server Configuration Profile>Import**, fill in the fields as required, and click **Preview**.

2.2.2 SCP import preview using RACADM:

To import preview a SCP file using RACADM, use a `racadm set` command. For more examples or help, execute the `racadm help set` command.

RACADM example:

```
C:\>racadm -r 192.168.0.120 -u root -p calvin set -f R740_scp_file.xml -t xml
-l 192.168.0.130:/nfs --preview
```

2.2.3 SCP import preview using Redfish:

To import preview a SCP profile using Redfish, execute a POST call on OEM action `EID_674_Manager.ImportSystemConfigurationPreview`.

Redfish POST example:

```
URI: /redfish/v1/Managers/iDRAC.Embedded.1/Actions/Oem/EID_674_Manager.
ImportSystemConfigurationPreview
Header: content-type application/json
Body: Body: {'ShareParameters': {'ShareType': 'NFS', 'ShareName': '/nfs',
'IPAddress': '192.168.0.130', 'FileName': 'R740_scp_file.xml'}}
```

2.3 Import server configuration profile template

After you edit the SCP file or it passes import preview, you can import the SCP file to apply the configuration changes.

NOTE: After executing an import operation using any supported iDRAC UI, a job ID will be returned which you can query to check the status and validate if the process completed or failed. If failed, check the iDRAC Lifecycle Logs for more details. For more information on how to check the job status, refer to the workflow section of this document.

2.3.1 SCP import using the iDRAC UI:

To import an SCP profile using the iDRAC UI, go to **Configuration>Server Configuration Profile>Import**, fill in the fields as needed, and click **Import**.

2.3.2 SCP import using RACADM:

To import SCP file using RACADM, use the `racadm set` command. For more examples or help, execute the `racadm help set` command.

RACADM example:

```
C:\>racadm -r 192.168.0.120 -u root -p calvin set -f R740_scp_file.xml -t xml
-l 192.168.0.130:/nfs
```

2.3.3 SCP import using Redfish:

To import SCP profile using Redfish, execute a POST call on OEM action `EID_674_Manager.ImportSystemConfiguration`.

```
Redfish POST example:
URI: /redfish/v1/Managers/iDRAC.Embedded.1/Actions/Oem/EID_674_Manager.
ImportSystemConfiguration
Header: content-type application/json
Body: Body: { 'ShareParameters': {'ShareType': 'NFS', 'ShareName': '/nfs',
'IPAddress': '192.168.0.130', 'Target': 'ALL', 'FileName':
'R740_scp_file.xml'}}
```

3 Server configuration profile end-to-end workflows

This section covers multiple end-to-end workflows using the server configuration profile feature. Different workflows use different iDRAC interfaces, stacking multiple configuration settings and performing operating system installation.

NOTE: All workflows discussed here are performed on a PowerEdge C6420 server with iDRAC9 version 4.00.

3.1 Workflow using the iDRAC UI

In this workflow example using the iDRAC UI, we describe the following process:

- SCP export in XML format to an NFS share, the same share which already contains the operating system ISO, and then query the job status to confirm that it is completed.
- Modify the SCP file to change iDRAC and NIC attributes, change LifecycleController attributes to set up attended ESXi operating system install.
- SCP import preview, and then query the job status to confirm that it is completed.
- SCP import using the same NFS share.
- Show server reboot and explain server and job execution behavior.
- Once configuration is complete, show the server booting to the attached ESXi ISO image.
- Confirm that the operating system installation process has started.
- Confirm that the import job status is marked completed, and show the boot to network ISO job status.

1. Export the current server configuration settings to an NFS share using the iDRAC UI. This NFS share already has the ESXi ISO image copied to it, simplifying the SCP import process. Launch the iDRAC UI and go to **Configuration>Server Configuration Profile>Export**. Fill in the necessary options to perform an export to the NFS share.

Table 4

Export options	**Settings**
Location Type	Network Share
Filename	Any unique name but for this workflow, will be passing in ‘C6420_scp_file.xml’
Protocol	NFS
IP Address	Pass in the NFS share IP
Share Name	Pass in NFS share name
Export Components	All
Export Type	Basic’
Export File Format	XML’

When all the fields have been filled in, click **Export**. A popup message is displayed stating that the job ID has been created. Click **Job Queue** to validate the job status.

2. Go to the NFS share and confirm that the SCP file exists.

```
[root@linux nfs]# pwd
/nfs
[root@linux nfs]# ls -la *.iso
-rw-r--r-- 1 root root   8658944 Aug 29  2011 boot.iso
```

```
-rwxr--r-- 1 root root 309999616 Feb  1  2019 esxi_5u1.iso
[root@linux nfs]# ls -la *.xml
-rw-rw-r-- 1 root 991 167814 Dec 20 12:46 C6420_scp_file.xml
[root@linux nfs]#
```

3. Using any editor, open the SCP file to make changes. In this workflow, we will change iDRAC and NIC attributes, adding a new iDRAC user, and setting Lifecycle Controller attributes to perform operating system installation.

Table 5 DRAC FQDD, iDRAC.Embedded.1 attribute settings

Attribute	Setting
Telnet.1#Enable	Enabled
Users.7#UserName	demo user
Users.7#Password	calvin
Users.7#Privilege	511
Users.7#Enable	Enabled

Table 6 NIC FQDD, NIC.Mezzanine.3-2-1 attribute settings

Attribute	Setting
LegacyBootProto	NONE
WakeOnLan	Enabled

Table 7 LC FQDD, LifecycleController.Embedded.1 attribute settings

Attribute	Setting
OSD.1#OSMediaShareType	local Since the ISO is in the same directory path as the SCP file, we don’t have to worry about filling in the other LC OSD attributes.
OSD.1#OSMediaName	esxi_5u1.iso
OSD.1#ExposeDuration	3600 This means the ISO will only be attached for 1 hour once the installation process starts. After 1 hour, the ISO is auto detached.

NOTE: Since we are installing ESXi, there is no need to unpack and attach a driver pack. In other workflows, we will show how to perform this operation for operating system installation.

Example of the edited XML SCP file which has been stripped down to only show the attributes we edited:

```
<SystemConfiguration>
<Component FQDD="NIC.Mezzanine.3-2-1">
 <Attribute Name="LegacyBootProto">NONE</Attribute>
 <Attribute Name="WakeOnLan">Enabled</Attribute>
</Component>
<Component FQDD="iDRAC.Embedded.1">
 <Attribute Name="Users.7#UserName">demo_user</Attribute>
 <Attribute Name="Users.7#Password">calvin</Attribute>
 <Attribute Name="Users.7#Privilege">511</Attribute>
 <Attribute Name="Users.7#Enable">Enabled</Attribute>
</Component>
<Component FQDD="LifecycleController.Embedded.1">
 <Attribute Name="OSD.1#OSMediaShareType">local</Attribute>
 <Attribute Name="OSD.1#OSMediaName">esxi_5u1.iso</Attribute>
<Attribute Name="OSD.1#ExposeDuration">3600</Attribute>
</Component>
</SystemConfiguration>
```

4. After editing the SCP file, execute the import preview to make sure the SCP file is correct and has no errors before importing. Go to **iDRAC GUI> Configuration>Server Configuration Profile** and click **Import**. Fill in these fields:

   - Location Type: Select “Network Share”
   - File Name: Pass in “C6420_scp_file.xml” for this workflow
   - Protocol: Select “NFS”
   - Ip Address: “Pass in NFS share IP”
   - Share Name: “Pass in NFS share name”
   - Import Components: Select “All”
5. Once all fields have been populated, click **Preview**. A popup message is displayed stating that a job ID has been created. Click **Job Queue** and validate the new entry, expand the **Import Preview** job ID, and confirm that a success message is displayed.
6. Go to **iDRAC GUI> Configuration>Server Configuration Profile** and click **Import**. Fill in the fields for Import Preview:
   - Location Type: Select “Network Share”
   - File Name: Pass in “C6420_scp_file.xml” for this workflow
   - Protocol: Select “NFS”
   - Ip Address: “Pass in NFS share IP”
   - Share Name: “Pass in NFS share name”
   - Import Components: Select “All”
7. When all fields have been populated, click **Import.** A popup message is displayed asking you to confirm initiation of the import operation. Click **OK**. Another popup message is displayed confirming that a job ID has been created. Click **Job Queue**, and verify that the status of your import job ID is **Running**.

   The server should automatically reboot. During POST, you will see a flag set stating, **Server Configuration Requested by Lifecycle Controller**. Once POST completes, the server will enter LC **Automated Task Application** to apply configuration changes. Once this process is complete, the server will reboot.

   After all configuration changes are applied and the server reboots, a flag will be set in POST to boot to the **Service Partition** or the attached ISO image. Once POST completes, you should see the operating system installation start.

NOTE: Since we selected an attended install, we will have to interact with the operating system to complete the process.

Once the server has booted to the attached ISO and started the installation process, the SCP import job is marked completed. Go back to the **iDRAC GUI> Maintenance> Job Queue** page and verify that the import job ID is marked completed.

You will also notice that a **Boot To Network ISO** job ID was created and marked successfully completed. This job ID is created during the SCP import job ID process to validate that the boot to network ISO process is ether successful or failed.

NOTE: The overall SCP import job will is not marked completed until the boot to network ISO job ID is marked completed or failed first.

8. Once the operating system installation is complete, if you need to detach the operating system drivers before the attach timeout has been reached, you can manually detach them using the iDRAC UI. Go to **Configuration> Virtual Media** and select the **Unmount Drivers** option to manually detach the operating system driver pack.

3.2 Workflow using the remote RACADM CLI

In this workflow using remote RACADM CLI, the following actions are performed:

- SCP export in JSON format to the CIFS share, the same share which already contains the operating system ISO and the kickstart file, and then query the job status to show completed.
- Modify the SCP file to change the BIOS settings:
  - Set the HD placeholder as the first device in the UEFI boot order.
  - Set the local key management (LKM) on the storage controller.
  - Create a RAID1 locked volume on which the operating systems is installed because of kickstart file settings.
  - Change Lifecycle Controller attributes to set up unattended RedHat (RHEL) 7.6 operating system install.
  - Unpack the driver pack.

NOTE: Setting HD placeholder as first device in the boot order, after the operating system installation is complete, the boot entry that UEFI creates for the operating system installation, this will now be the first device in the boot order.

- SCP preview, validate no issues with the SCP file.
- Show example of SCP import preview failure with viewing the job ID config results explaining the failure.
- SCP import using the same CIFS share.
- Query import job status until marked completed.
- Workflow complete.

NOTE: If needed, refer to the UI workflow section which shows screenshots of server behavior once the import job has been created and the server reboots.

1. Export the current server configuration settings to a CIFS share using remote RACADM `get` command. This CIFS share already contains the RHEL 7.6 ISO and kickstart file.

   If the `get` command is successful, a job ID is returned. Query the job ID using a `racadm jobqueue view` command to confirm that it is marked completed successfully.

   Note: For more details on supported parameters and examples for get command, execute `racadm help get`.

   ```
   C:\>racadm -r 100.65.84.70 -u root -p calvin --nocertwarn get -f
   C6420_scp_file.json -t json -u administrator -p P@ssw0rd -l
   //100.65.84.72/cifs_share_vm
   RAC976: Export configuration XML file operation initiated.
   Use the "racadm jobqueue view -i JID_768692922198" command to view the status
   of the operation.
   ```

```
C:\>racadm -r 100.65.84.70 -u root -p calvin --nocertwarn jobqueue view -i
JID_768692922198
---------------------------- JOB -------------------------
[Job ID=JID_768692922198]
Job Name=Export: Server Configuration Profile
Status=Running
Start Time=[Not Applicable]
Expiration Time=[Not Applicable]
Message=[SYS057: Exporting Server Configuration Profile.]
Percent Complete=[60]
----------------------------------------------------------
C:\>racadm -r 100.65.84.70 -u root -p calvin --nocertwarn jobqueue view -i
JID_768692922198
---------------------------- JOB -------------------------
[Job ID=JID_768692922198]
Job Name=Export: Server Configuration Profile
Status=Completed
Start Time=[Not Applicable]
Expiration Time=[Not Applicable]
Message=[SYS043: Successfully exported Server Configuration Profile]
Percent Complete=[100]
----------------------------------------------------------
```

2. Go to the CIFS share and validate the SCP JSON file was created. You will also notice the RHEL 7.6 ISO and kickstart file (ks.cfg) are also in the same share path.
3. Using any editor, open the SCP file to make changes. For this workflow, I'll be changing BIOS settings, set HD placeholder as first device in the UEFI boot order, set local key management (LKM) on the storage controller, create RAID 1 locked volume which the operating system will be installed on because of kickstart file settings, change Lifecycle Controller attributes to set up unattended RedHat (RHEL) 7.6 operating system install, unpack driver pack.

Table 8 RAID FQDD, RAID.Mezzanine.1-1 attribute settings

Attribute	Setting
RAIDresetConfig	True
EncryptionMode	Local Key Management
KeyID	Testkey
NewControllerKey	Newkey12##

Table 9 RAID VD FQDD, Disk.Virtual.0:RAID.Mezzanine.1-1 attribute settings

Attribute	Setting
RAIDaction	Created
LockStatus	Locked
Name	RHEL7 RAID1
Size	0
StripeSize	128
SpanDepth	1
SpanLength	2
RAIDTypes	RAID1
IncludedPhysicalDiskID	Disk.Bay.0:Enclosure.Internal.0-1:RAID.Mezzanine.1-1
IncludedPhysicalDiskID	Disk.Bay.2:Enclosure.Internal.0-1:RAID.Mezzanine.1-1

NOTE: "IncludedPhysicalDiskID" attribute needs to be passed per disk you will be needing to create the RAID volume.

Table 10 BIOS FQDD, BIOS.Setup.1-1 attribute settings

Attribute	**Setting**
MemTest	Enabled
UefiBootSeq	RAID.Mezzanine.1-1

Table 11 LC FQDD, LifecycleController.Embedded.1 attribute settings

Attribute	**Setting**
OSD.1#OSName	Red Hat Enterprise Linux 7.6 x64
OSD.1#OSMediaShareType	local
OSD.1#OSMediaName	RHEL_7_6_Server_x86_64.iso
OSD.1#AnswerFileName	ks.cfg
OSD.1#ExposeDuration	7200

The “OSD.1#OSName” value comes from attribute "OSD.1#SupportedOSList". Passing in this value means iDRAC will unpack and attach the driver pack before attaching the ISO and performing the operating system installation.

Since the ISO is in the same directory path as the SCP file, OSMediaShareType can be set to local and we don’t have to worry about filling in the other network OSD attributes.

ExposeDuration is set to 7200, which means the ISO will only be attached for 2 hours once the installation process starts. After 2 hours, the ISO will get auto detached.

NOTE: When using JSON SCP file, you need to make sure for each attribute you want to configure, "Set On Import" is set to “True”.

Here is an example of the edited SCP JSON file which has been stripped down to only show the attributes that are edited:

```
{ "SystemConfiguration": {
  "Components": [
  { "FQDD": "RAID.Mezzanine.1-1",
    "Attributes": [
    { "Name": "RAIDresetConfig",
      "Value": "True",
      "Set On Import": "True",
      "Comment": "Read and Write" },
    { "Name": "EncryptionMode",
      "Value": "Local Key Management",
      "Set On Import": "True",
      "Comment": "Read and Write" },
    { "Name": "KeyID",
      "Value": "testkey",
      "Set On Import": "True",
      "Comment": "Read and Write" },
    { "Name": "NewControllerKey",
      "Value": "Newkey12##",
      "Set On Import": "True",
      "Comment": "Read and Write" }
    ],
    "Components": [
    { "FQDD": "Disk.Virtual.0:RAID.Mezzanine.1-1",
      "Attributes": [
      { "Name": "RAIDaction",
        "Value": "Create",
        "Set On Import": "True",
        "Comment": "Read and Write" },
      { "Name": "LockStatus",
        "Value": "Locked",
        "Set On Import": "True",
        "Comment": "Read and Write" },
      { "Name": "Name",
        "Value": "RHEL7 RAID1",
        "Set On Import": "True",
        "Comment": "Read and Write" },
      { "Name": "Size",
        "Value": "0",
        "Set On Import": "True",
        "Comment": "Read and Write" },
      { "Name": "StripeSize",
        "Value": "128",
        "Set On Import": "True",
        "Comment": "Read and Write" },
      { "Name": "SpanDepth",
        "Value": "1",
        "Set On Import": "True",
        "Comment": "Read and Write" },
      { "Name": "SpanLength",
        "Value": "2",
        "Set On Import": "True",
        "Comment": "Read and Write" },
      { "Name": "RAIDTypes",
        "Value": "RAID 1",
        "Set On Import": "True",
        "Comment": "Read and Write" },
      { "Name": "IncludedPhysicalDiskID",
        "Value": "Disk.Bay.0:Enclosure.Internal.0-1:RAID.Mezzanine.1-1",
        "Set On Import": "True",
        "Comment": "Read and Write" },
      { "Name": "IncludedPhysicalDiskID",
        "Value": "Disk.Bay.2:Enclosure.Internal.0-1:RAID.Mezzanine.1-1",
        "Set On Import": "True",
        "Comment": "Read and Write" }
        ]}
    ]},
  { "FQDD": "BIOS.Setup.1-1",
    "Attributes": [
    { "Name": "MemTest",
      "Value": "Enabled",
      "Set On Import": "True",
      "Comment": "Read and Write" },
    { "Name": "UefiBootSeq",
      "Value": "RAID.Mezzanine.1-1",
      "Set On Import": "True",
      "Comment": "Read and Write" }
 ]},
      { "FQDD": "LifecycleController.Embedded.1",
    "Attributes": [
    { "Name": "OSD.1#OSName",
      "Value": "Red Hat Enterprise Linux 7.6 x64",
      "Set On Import": "True",
      "Comment": "Read and Write" },
    { "Name": "OSD.1#OSMediaShareType",
      "Value": "local",
      "Set On Import": "True",
      "Comment": "Read and Write" },
    { "Name": "OSD.1#OSMediaName",
      "Value": "RHEL_7_6_Server_x86_64.iso",
      "Set On Import": "True",
      "Comment": "Read and Write" },
    { "Name": "OSD.1#AnswerFileName",
      "Value": "ks.cfg",
      "Set On Import": "True",
      "Comment": "Read and Write" },
    { "Name": "OSD.1#ExposeDuration",
      "Value": "7200",
      "Set On Import": "True",
      "Comment": "Read and Write" }
    ]}
]}
}
```

4. After editing the SCP file, execute an import preview to make sure the SCP file is correct and has no errors before importing. To perform SCP import preview, we will be executing “racadm set” command passing in “—preview” parameter.

   If import preview command passes, a job ID will be returned. Take this job ID and query to make sure it returns completed successfully message.

```
C:\>racadm -r 100.65.84.70 -u root -p calvin --nocertwarn set -f
C6420_scp_file.json -t json -u administrator -p P@ssw0rd -l
//100.65.84.72/cifs_share_vm --preview
RAC1114 : Configuration XML preview operation job is initiated.
      Monitor the status of the preview operation job by running the
       racadm command "racadm jobqueue view -i JID_768724066327".
C:\>racadm -r 100.65.84.70 -u root -p calvin --nocertwarn jobqueue view -i
JID_768724066327
---------------------------- JOB -------------------------
```

```
[Job ID=JID_768724066327]
Job Name=Preview Configuration
Status=Completed
Start Time=[Not Applicable]
Expiration Time=[Not Applicable]
Message=[SYS081: Successfully previewed Server Configuration Profile import
operation.]
Percent Complete=[100]
----------------------------------------------------------
```

NOTE: If you get any type of job failure, you can view the job ID config results which will tell you the details of why preview failed.

The example that follows shows that the preview job failed. To access detailed information on the failure, execute the “lclog viewconfigresult” command on the import preview job ID.

```
C:\>racadm -r 100.65.84.70 -u root -p calvin --nocertwarn jobqueue view -i
JID_768723151500
---------------------------- JOB -------------------------
[Job ID=JID_768723151500]
Job Name=Preview Configuration
Status=Failed
Start Time=[Not Applicable]
Expiration Time=[Not Applicable]
Message=[SYS079: The Preview operation indicates the input file for Server
Configuration Profile is not compliant with the configuration profile
schema.]
Percent Complete=[100]
----------------------------------------------------------
C:\>racadm -r 100.65.84.70 -u root -p calvin --nocertwarn lclog
viewconfigresult -j JID_768723151500
SeqNumber       = 2708
Job Name        = Import Configuration
Message ID      = SYS048
Message         = Server Configuration Profile input file contains invalid
characters, ( after } expected } ] or , ) at line 79
```

5. After the validated import preview has passed successfully, import this SCP file to apply configuration changes and perform an unattended operating system installation.

   Note: Use the exact same syntax as Step 4 for the “set” command, except that you do not pass in the “—preview” option.

   ```
   C:\>racadm -r 100.65.84.70 -u root -p calvin --nocertwarn set -f
   C6420_scp_file.json -t json -u administrator -p P@ssw0rd -l
   //100.65.84.72/cifs_share_vm
   RAC977: Import configuration XML file operation initiated.
   Use the "racadm jobqueue view -i JID_779832077111" command to view the status
   of the operation.
   ```

6. The Job ID will be returned to the screen. Query the job ID until the job ID status is completed successfully. After the job ID is marked completed successfully, unattended operating system installation will start.

```
C:\>racadm -r 100.65.84.70 -u root -p calvin --nocertwarn jobqueue view -i
JID_779832077111
---------------------------- JOB -------------------------
[Job ID=JID_779832077111]
Job Name=Configure: Import Server Configuration Profile
Status=Running
Start Time=[Not Applicable]
Expiration Time=[Not Applicable]
Message=[SYS058: Applying configuration changes.]
Percent Complete=[20]
----------------------------------------------------------
C:\>racadm -r 100.65.84.70 -u root -p calvin --nocertwarn jobqueue view -i
JID_779832077111
---------------------------- JOB -------------------------
[Job ID=JID_779832077111]
Job Name=Configure: Import Server Configuration Profile
Status=Running
Start Time=[Not Applicable]
Expiration Time=[Not Applicable]
Message=[SYS032: Staged component configuration is complete.]
Percent Complete=[99]
----------------------------------------------------------
C:\>racadm -r 100.65.84.70 -u root -p calvin --nocertwarn jobqueue view -i
JID_779832077111
---------------------------- JOB -------------------------
[Job ID=JID_779832077111]
Job Name=Configure: Import Server Configuration Profile
Status=Completed
Start Time=[Not Applicable]
Expiration Time=[Not Applicable]
Message=[SYS053: Successfully imported and applied Server Configuration
Profile.]
Percent Complete=[100]
----------------------------------------------------------
```

7. After operating system installation is complete, if you need to detach the operating system drivers before the attach timeout has been reached, you can manually detach them using the “racadm driverpack detach” command.

```
C:\>racadm -r 100.65.84.70 -u root -p calvin --nocertwarn driverpack detach
RAC1247: The driverpack detach operation is successfully completed.
```

3.3 Workflow using Redfish

In this workflow using Redfish, the following actions are performed:

- SCP selective (get only Lifecycle Controller attributes) export (XML format) to HTTP share. The operating system ISO will be located on a CIFS share.
- Modify the SCP file to change Lifecycle Controller attributes to set up attended Windows Server 2019 operating system install, unpack driver pack.
- SCP preview, validate no issues with the SCP file.

- SCP import using the same HTTP share.
- Query import job status until marked completed.
- Workflow complete

Note: If needed, refer to the GUI workflow section which shows screenshots of server behavior once the import job has been created and the server reboots.

1. Perform selective export in XML format, exporting only Lifecycle Controller attributes to an HTTP share using Redfish.

   Note: For this Redfish workflow, use Postman to make Redfish calls to the iDRAC.

   SCP export executes a POST action which requires:

   Table 12 POST action settings

Setting	Value
Action	“Oem/EID_674_Manager.ExportSystemConfiguration”
Header	“Content_type application/json”
Body Parameters	“ExportFormat” and “SharedParameters”
Authorization	“Basic Auth” (passing in iDRAC username / password)

   In the response output, you get a status code of 202 for success. For the “Location” value, this is a URI which contains the job ID.

2. Take the location URI and execute a GET to check the job status. You should get back a status code of 200. In the output, you should see a message string stating that the export was successful.
3. Next, modify the SCP file on the HTTP share. In the SCP file, only Lifecycle Controller attributes are available since we performed selective export. Because the operating system ISO is located on a different network share, you must modify additional Lifecycle Controller attributes to inform the iDRAC where the ISO is located. The following example shows the SCP file modified where the location of the operating system ISO is passed in on the CIFS share.

```
<SystemConfiguration>
<Component FQDD="LifecycleController.Embedded.1">
 <!-- <Attribute Name="OSD.1#SupportedOSList">Microsoft Windows Server 2016,
Microsoft Windows Server 2012 R2, Microsoft Windows Server 2019, Red Hat
Enterprise Linux 8.0 x64, Red Hat Enterprise Linux 7.6 x64, SuSE Enterprise
Linux 15 x64</Attribute> -->
 <Attribute Name="OSD.1#OSName">Microsoft Windows Server 2019</Attribute>
 <Attribute Name="OSD.1#OSMediaShareIP">100.65.84.72</Attribute>
 <Attribute Name="OSD.1#OSMediaShareName">cifs_share_vm</Attribute>
 <Attribute Name="OSD.1#OSMediaShareUsername">administrator</Attribute>
 <Attribute Name="OSD.1#OSMediaSharePassword">P@ssw0rd</Attribute>
 <Attribute Name="OSD.1#OSMediaShareType">CIFS</Attribute>
 <Attribute Name="OSD.1#OSMediaName">Windows Svrs 2019 English
Std.iso</Attribute>
</Component>
</SystemConfiguration>
```

4. After the SCP file has been edited, execute the import preview using action “EID_674_Manager.ImportSystemConfigurationPreview”.

   SCP import preview will be executing POST action which requires:

Table 13 POST action settings

Setting	Value
Action	“Oem/EID_674_Manager.ImportSystemConfigurationPreview”
Header	“Content_type application/json”
Body Parameters	“SharedParameters”
Authorization	“Basic Auth” (passing in iDRAC username / password)

After you execute a POST command, a status code of 202 is returned along with a job ID URI value for the Location key.

5. Take the location URI and execute a GET to check the job status. You should get back a status code of 200. In the output, you should see a message string reporting that the import preview was successful.
6. After validating that the SCP preview was successful, execute an import using action “EID_674_Manager.ImportSystemConfiguration”.

   SCP import preview will be using POST action which requires:

Table 14 POST action settings

Setting	Value
Action	“Oem/EID_674_Manager.ImportSystemConfiguration”
Header	“Content_type application/json”
Body Parameters	“SharedParameters”
Authorization	“Basic Auth” (passing in iDRAC username / password)

After you execute POST command, a status code of 202 is returned along with job ID URI value for Location key.

7. Take the location URI and execute a GET to check the job status. You should get back a status code of 200. In the output, you should see a message string stating importing server configuration profile. Continue to execute the GET command until you see a status of successfully completed.
8. The server should already have booted to the ISO. Since we performed an attended install, interact with the operating system installation process to complete the installation.

NOTE: once the operating system install is complete, if you want to manually detach the driver pack, execute POST action “DellOSDeploymentService.DetachDrivers”.

Detach operating system drivers will be using POST action which requires:

Table 15 POST action settings

Setting	Value
Action	“DellOSDeploymentService.DetachDrivers”
Header	“Content_type application/json”
Body Parameters	Leave empty
Authorization	“Basic Auth” (passing in iDRAC username / password)

After you execute the POST command, a status code of 200 is returned along with success message.

4 Troubleshooting

The following sections detail some of the most common issues seen while performing operating system install via SCP and how to resolve them

4.1 Operating system driver pack issues

The following are symptoms of issues with the operating system driver pack:

- If the below is seen in the SCP template file for the OSD field:

```
<!-- <Attribute Name="OSD.1#SupportedOSList"></Attribute> -->
 <Attribute Name="OSD.1#OSName"></Attribute>
 <Attribute Name="OSD.1#OSMediaShareIP"></Attribute>
 <Attribute Name="OSD.1#OSMediaShareName"></Attribute>
 <Attribute Name="OSD.1#OSMediaShareUsername"></Attribute>
 <Attribute Name="OSD.1#OSMediaSharePassword">******</Attribute>
 <Attribute Name="OSD.1#OSMediaShareDomainName"></Attribute>
 <Attribute Name="OSD.1#OSMediaShareType"></Attribute>
 <Attribute Name="OSD.1#OSMediaName"></Attribute>
 <Attribute Name="OSD.1#AnswerFileName"></Attribute>
 <Attribute Name="OSD.1#ExposeDuration"></Attribute>
 <Attribute Name="OSD.1#OSMediaHashType"></Attribute>
 <Attribute Name="OSD.1#OSMediaHashValue"></Attribute>
```

During the SCP import, the "OSD: UnpackAndAttach" job fails

Either of these means that the latest drivers for the system have not been installed or are not recognized by the system. To resolve this issue, do the following:

1. Go to dell.com/support and look up your server either by model or service tag.
2. Go to the **Drivers and Downloads** tab and select **Drivers for OS Deployment** from the **Category** dropdown box.
3. Select the latest driver pack file and download and install it on your server. This can be done via the iDRAC UI by going to **Maintenance>System Update** and uploading the driver pack file.
4. After installing the driver pack, retry the operating system install operation.

4.2 Network issues

The following are symptoms of network share connection issues:

If network connectivity issues are a problem, the SCP import job will fail as seen below:

```
racadm jobqueue view
----------------------------------------------------------
[Job ID=JID_xxxxx]
Job Name=Configure: Import Server Configuration Profile
Status=Failed
Start Time=[Not Applicable]
Expiration Time=[Not Applicable]
```

```
Message=[SYS067: Unable to complete application of configuration profile
values.]
Percent Complete=[100]
----------------------------------------------------------
To get more information on why the job failed, execute the following command:
racadm lclog viewconfigresult -j JID_xxxxx
SeqNumber       = 148
FQDD            = LifecycleController.Embedded.1
Job Name        = Import Configuration
DisplayValue    = OS Media Share IP Address
Name            = OSD.1#OSMediaShareIP
OldValue        = ""
NewValue        = 10.36.99.99
Status          = Failure
MessageID       = OSD16
ErrCode         = 10366
```

This shows which value is causing the error. Double check the network share connection info and credentials and retry the command.

4.3 Corrupt operating system installation file

If the operating system installation file, the ISO file for example, is corrupt, three jobs will be created and completed successfully just as they would be if the installation file is good:

```
racadm jobqueue view
-------------------------JOB QUEUE------------------------
[Job ID=JID_783469298257]
Job Name=Configure: Import Server Configuration Profile
Status=Completed
Start Time=[Not Applicable]
Expiration Time=[Not Applicable]
Message=[SYS053: Successfully imported and applied Server Configuration
Profile.]
Percent Complete=[100]
----------------------------------------------------------
[Job ID=JID_783469403822]
Job Name=OSD: UnpackAndAttach
Status=Completed
Start Time=[Not Applicable]
Expiration Time=[Not Applicable]
Message=[OSD1: The command was successful.]
Percent Complete=[100]
----------------------------------------------------------
[Job ID=JID_783469655226]
Job Name=OSD: BootTONetworkISO
Status=Completed
Start Time=[Not Applicable]
Expiration Time=[Not Applicable]
Message=[OSD1: The command was successful.]
Percent Complete=[100]
```

```
----------------------------------------------------------
```

However, once the system tries to boot to the corrupt media an error will be seen on the host.

```
Booting from vFlash Media ISOIMG
Boot Failed: vFlash Media ISOIMG

Boot Failed:
Please ensure a compatible bootable media is available.

Available Actions:
F1 to Continue and Retry Boot Order
F2 for System Setup (BIOS)
F11 for Boot Manager
```

If this happens, make sure that the operating system install file is good and retry the SCP operation.

5 Useful Links

- *Using Server Configuration Profiles (SCP) on PowerEdge Servers*: https://downloads.dell.com/solutions/dell-management-solution-resources/ServerCloning_SCP%20v2_50%28DTC%20copy%29.pdf
- SCP .xml file structure whitepaper: https://downloads.dell.com/solutions/general-solution-resources/White%20Papers/Server%20Configuration%20XML%20File.pdf

6 Additional notes

- The SCP option for *end host power state* will be overwritten when OSD operations are provided in the template.
- The operating system install process might require manual interaction via a **Press Any Key** prompt if the host already has an operating system installed.
- The operating system name value you pass in the SCP file must not contain any whitespace characters. For example, *Windows Server 2019.iso* is an invalid value, but *Windows_Server_2019.iso* is a valid value.
- SCP import/preview operations are blocked while the driver pack is still exposed or attached.
- The SCP import operation ignores the driver pack attach request when the operating system media information fails the network connectivity check. This is done as a convenience to allow the user to resolve any network or template issues before blocking Lifecycle Controller operations with the driver pack.
- Unattended ESXi install using SCP has a limitation. The ESXi ISO itself is not set up to know where to look for the kickstart file compared to RHEL or SUSE. To perform an unattended ESXi install using SCP, you must modify the ISO to ensure the ISO can locate the kickstart file on the network share. In the SCP file itself, you only need to pass in the ESXi ISO name.