



# Configure Cisco UCS PowerTool Suite

This chapter consists of the following sections:

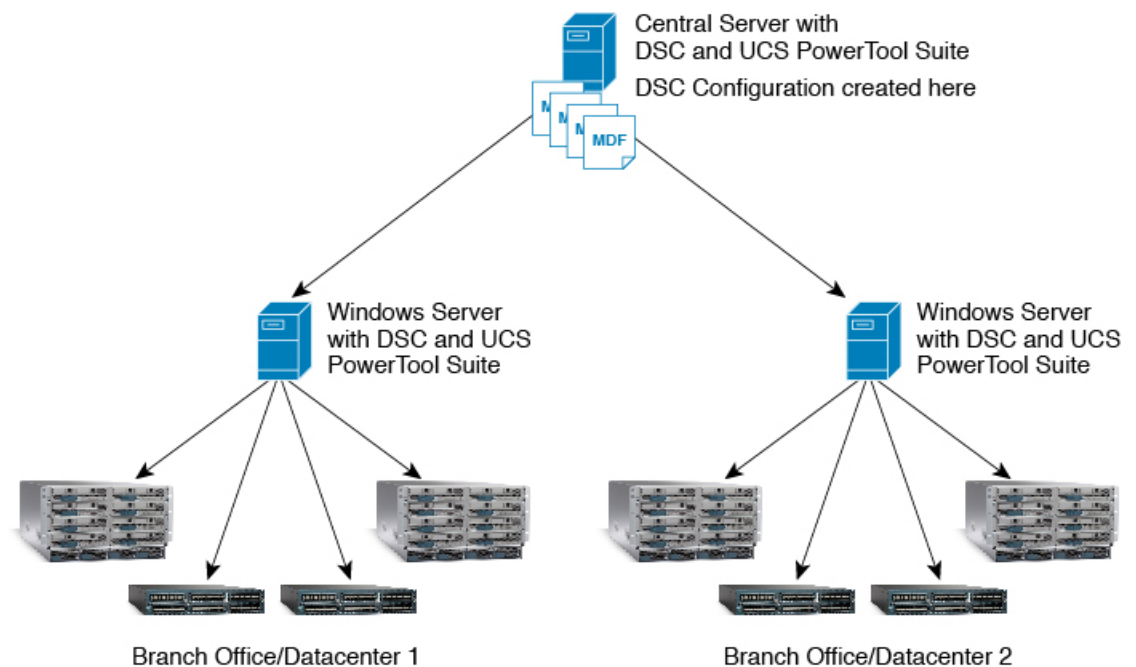
- [Cisco UCS Desired State Configuration \(DSC\) Resource, on page 1](#)
- [Cisco UCS Core Module, on page 2](#)
- [Getting Cisco UCS Software Images, on page 2](#)
- [Cisco UCS Hardware and Software Compatibility Tool Integration, on page 3](#)

## Cisco UCS Desired State Configuration (DSC) Resource

Desired State Configuration (DSC) is a management platform in Windows PowerShell which enables you to configure, deploy, and manage systems.

Cisco UCS DSC resource enables you to configure Cisco UCS Manager, and Cisco IMC using the Windows PowerShell DSC management platform.

**Figure 1: Cisco UCS DSC Solution Architecture**



To apply any configuration using Cisco UCS DSC resource, install Cisco UCS PowerTool Suite on Windows server with DSC.

The Cisco UCS DSC architecture is comprised of the following:

- **Central Server**—This server is used to write the UCS DSC configuration scripts
- **Intermediate Server**— Central server deploys the configuration to this server. This server applies the configuration to the Cisco UCS Manager, or Cisco IMC

## Cisco UCS Core Module

Cisco UCS core module contains cmdlets for features or actions common to all Cisco UCS modules. A cmdlet replaces the duplicate cmdlets available in the older releases of Cisco UCS Manager, Cisco IMC and Cisco UCS Central PowerTool. The cmdlets in **Retained Cmdlet** column have been retained and it works for all three PowerTool modules.

Retained Cmdlet	Cmdlet Removed in Cisco Central	Cmdlet Removed in Cisco IMC
<b>Get-UcsPowerToolCommunity</b>	Get-UcsCentralPowerToolCommunity	Get-ImcPowerToolCommunity
<b>Get-UcsPowerToolConfiguration</b>	Get-UcsCentralPowerToolConfiguration	Get-ImcPowerToolConfiguration
<b>Set-UcsPowerToolConfiguration</b>	Set-UcsCentralPowerToolConfiguration	Set-ImcPowerToolConfiguration
<b>Get-UcsPSSession</b>	Get-UcsCentralPSSession	Get-ImcPSSession
<b>Export-UcsPSSession</b>	Export-UcsCentralPSSession	Export-ImcPSSession

If you are using cmdlets that are removed from this release in any of the scripts, then use the cmdlets from the **Retained Cmdlet** column instead.

## Getting Cisco UCS Software Images

The following cmdlets can be used to get the available driver or firmware images for Cisco UCS Manager, Cisco UCS Central, or Cisco IMC servers from Cisco.com.

- **Get-UcsSoftwareImageList**—To get all the images available for either UCS Manager, Cisco IMC, or UCS Central
- **Get-UcsSoftwareImage**—This cmdlet downloads the image to a local file system. This cmdlet takes pipeline input from the **Get-UcsSoftwareImageList** cmdlet.

### Get-UcsSoftwareImageList

#### Syntax

```
Get-UcsSoftwareImageList -Credential <PSCredential>
[-Category <string>] [-Type <string>] [-Model <string>] [-AllReleases]
[-Proxy <WebProxy>] [-Ucs <BaseHandle[]>] [<CommonParameters>]
```

```
Get-UcsSoftwareImageList -Credential <PSCredential>
```

```
-MdfId <uint> -SoftwareId <uint> [-AllReleases] [-Proxy <WebProxy>]
[-Ucs <BaseHandle[]>] [<CommonParameters>]
```



**Note** *AllReleases* switch parameter is available from UCS PowerTool Suite 2.0(2) and higher. Specifying the *-AllRelease* switch parameter, lists all the available images in Cisco.com for the selected device type.

*-Version* parameter is valid for UCS PowerTool Suite Release 2.0(1) only.

## Get-UcsSoftwareImage

### Syntax

```
Get-UcsSoftwareImage -SoftwareImage <ImageDetails> -Path <string> [-EulaAccepted] [-Xml]
[<CommonParameters>]
```

### Example

```
Get-UcsSoftwareImageList
Get-UcsSoftwareImageList -Category Default
Get-UcsSoftwareImageList -Category ModularServer
Get-UcsSoftwareImageList -Category ModularServer -Type Firmware
Get-UcsSoftwareImageList -Category ModularServer -Type Drivers
Get-UcsSoftwareImageList -Type Firmware
Get-UcsSoftwareImageList -Type Drivers
$images= Get-UcsSoftwareImageList
$image[0] | Get-UcsSoftwareImage -Path "C:\Images" -EulaAccepted -verbose
Get-UcsSoftwareImageList | Get-UcsSoftwareImage -Path "C:\Images" -verbose
```

The following cmdlets which were used for getting software images from Cisco.com are deprecated from this release:

- Get-UcsCcoImageList
- Get-UcsCcoImage

# Cisco UCS Hardware and Software Compatibility Tool Integration

Cisco UCS Hardware and Software Compatibility List (HCL) Tool provides interoperability information for UCS components and configurations tested and validated by Cisco, by Cisco partners, or both.

From Release 2.1.1 of UCS PowerTool Suite, custom cmdlets for interacting with the HCL tool are added. With this feature, you can create a Hardware Profile from a UCS blade or rack server and upload it to the HCL tool. Also, you can get the recommended versions of the firmware or driver versions from the HCL tool by selecting the uploaded hardware profile.

## Connect to Cisco UCS

Before executing any cmdlet, you should first connect to Cisco UCS using the following:

```
C:\> $cred = Get-Credential
C:\> $ucs = Connect-Ucs -Name <<IP Address>> -Credential $cred
```

## Get-UcsOsVendor

The Get-UcsOsVendor cmdlet is used to get the operating system vendor code and name from the HCL tool.

### Syntax

```
Get-UcsOsVendor [-Id <string>] [-Proxy <WebProxy>] [<CommonParameters>]
```

### Detailed Description

Property	Description	Required
Id	Specifies the OsVendor Id.	False
Proxy	Specifies the web proxy to be used for the communication with HCL api.	False
CommonParameters	This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable.	

### Example

```
C:\> $osVendor = Get-UcsOsVendor
```

```
C:\> $OSVendor
```

```
T_ID ID OSVENDOR
---- -- -
2    0 Citrix
3    1 Microsoft
6    2 SuSE
4    3 Oracle
5    4 Red Hat
8    5 VMware
7    7 Ubuntu
1    8 CentOS
```

## Get-UcsOperatingSystem

The Get-UcsOperatingSystem cmdlet is used to get all the supported operating systems for the selected vendor.

### Syntax

```
Parameter Set: Vendor
```

```
Get-UcsOperatingSystem -OsVendor <OsVendor> [-Proxy <WebProxy>] [<CommonParameters>]
```

```
Parameter Set: Id
```

```
Get-UcsOperatingSystem -OsVendorTreeId <string> [-Proxy <WebProxy>] [<CommonParameters>]
```

### Detailed Description

Property	Description	Required
OsVendor	Specifies the OsVendor object.	True
OsVendorTreeId	Specifies the tree id of OsVendor.	True

Property	Description	Required
Proxy	Specifies the web proxy to be used for the communication with HCL api.	False
CommonParameters	This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable.	

**Example**

The following example has Microsoft OS vendor details passed as `$osVendor [1]`.

```
C:\> $osVersion = Get-UcsOperatingSystem -OsVendor $osVendor[1]

C:\> $osVersion

T_ID ID OSVERSION
---- -- -
39 141 Windows Server 2008 R2 SP1
41 72 Windows Server 2012
38 338 Windows Server 2008 R2
42 118 Windows Server 2012 R2
40 175 Windows Server 2008 SP2
44 810 Windows Server 2019
43 601 Windows Server 2016
```

**New-UcsHardwareProfile**

The New-UcsHardwareProfile cmdlet is used to create a hardware profile from the selected server. The hardware profile created using this cmdlet can be uploaded to the HCL tool using the Add-UcsHardwareProfile cmdlet. This cmdlet takes managed object of the type ComputeBlade, ComputeRackUnit, ComputeServerNode, and ServiceProfile as pipeline input. By default, the cmdlet collects the serial number of the server. If you do not want the cmdlet to collect the serial number, use the `-NoSerialNumber` switch parameter.

**Syntax**

```
New-UcsHardwareProfile [-NamePrefix <string>] [-NoSerialNumber] -ManagedObject <ManagedObject>
-OsVendorCode <string> -OsVersionCode <string> [-Ucs <BaseHandle[]>] [<CommonParameters>]
New-UcsHardwareProfile -Name <string> [-NoPostFix] [-NoSerialNumber] -ManagedObject
<ManagedObject> -OsVendorCode <string> -OsVersionCode <string> [-Ucs <BaseHandle[]>]
[<CommonParameters>]
```

**Detailed Description**

Property	Description	Required
Name	Specifies the name of the Hardware profile.	True
NamePrefix	Specifies the name prefix of the Hardware profile.	False

Property	Description	Required
Path	Specifies path at which the Hardware Profile JSON file is located.	True
LiteralPath	Specifies path at which the Hardware Profile JSON file is located.	True
NoPostFix	Specifies that the cmdlet does not attach the postfix string to the Hardware Profile name.	False
ManagedObject	Specifies the managed object. You can pass an associated ServiceProfile or blade or RackUnit or ServerNode as the parameter or from pipeline	True
NoSerialNumber	Switch parameter instructs cmdlet not to capture Serial Number inside the Hardware Profile.	False
OsVendorCode	Specifies the operating system's vendor code obtained using Get-UcsOsVendor cmdlet.	True
OsVersionCode	Specifies the operating system code obtained using Get-UcsOperatingSystem cmdlet.	True
Ucs	Specifies the ucs handle or handles	False
CommonParameters	This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable.	

### Example

```
C:\> $managedObject = Get-UcsManagedObject -Dn "sys/rack-unit-1"

C:\> $hardwareProfile = New-UcsHardwareProfile -Name "Test-HardwareProfile"
                    -OsVendorCode $osVendor[1].T_ID -OsVersionCode $osVersion[1].T_ID -ManagedObject
                    $managedObject

C:\> $hardwareProfile | ConvertTo-Json -Depth 100 | Out-File C:\hardwareProfile.json
```

### Add-UcsHardwareProfile

The Add-UcsHardwareProfile cmdlet is used to upload the hardware profile to the HCL tool. Specify the Cisco.com credentials and hardware profile objects which were created using the New-UcsHardwareProfile cmdlet. You can save multiple hardware profiles for the account specified. You can also specify the JSON

file as input for uploading a hardware profile to the HCL tool. These hardware profiles can be used later to get the information about the recommended driver and firmware versions for the adaptors.

**Syntax**

```
Add-UcsHardwareProfile -Credential <PSCredential>
-HardwareProfile <CiscoHardwareProfile>
[-Proxy <WebProxy>] [-Xml] [<CommonParameters>]
```

```
Add-UcsHardwareProfile -Credential <pscredential>
-Path <string> [-Proxy <WebProxy>]
[-Xml] [<CommonParameters>]
```

```
Add-UcsHardwareProfile -Credential <pscredential>
-LiteralPath <string> [-Proxy <WebProxy>] [-Xml]
 [<CommonParameters>]
```

**Detailed Description**

Property	Description	Required
HardwareProfile	Specifies the hardware profile object created using the New-UcsHardwareProfile cmdlet.	True
Proxy	Specifies the web proxy to be used for the communication with HCL api.	False
Credential	Specifies the Cisco.com credential of user.	True
Path	Specifies path at which Hardware Profile JSON file is located	True
LiteralPath	Specifies path at which Hardware Profile JSON file is located.	True
CommonParameters	This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable.	

**Example**

```
C:\> $result = Add-UcsHardwareProfile -Credential $cecCred -Path C:\hardwareProfile.json

C:\> $result

CreationDate : 2019-06-20T06:37:07+00:00
LastModified : 2019-06-20T06:37:07+00:00
CecId       :
Id          : 9463
Name       : Test-HardwareProfile_0
UcsServer  : Cisco.Ucs.Core.CiscoUcsServer
```

## Get-UcsHardwareProfile

The Get-UcsHardwareProfile cmdlet is used to get all the hardware profiles present in the HCL tool for the given cisco.com credential. You can use the optional name or Id parameter to get the specific hardware profile.

### Syntax

```
Parameter Set: Default
Get-UcsHardwareProfile -Credential <PSCredential> [-Proxy <WebProxy>] [<CommonParameters>]

Parameter Set: Name
Get-UcsHardwareProfile -Credential <PSCredential> -Name <string> [-Proxy <WebProxy>]
[<CommonParameters>]

Parameter Set: Id
Get-UcsHardwareProfile -Credential <PSCredential> -Id <string> [-Proxy <WebProxy>]
[<CommonParameters>]
```

### Detailed Description

Property	Description	Required
Name	Specifies the name of the Hardware profile.	True
Id	Specifies the name of the Hardware Profile.	True
Proxy	Specifies the web proxy to be used for the communication with HCL api.	False
Credential	Specifies the Cisco.com credential of user.	True
CommonParameters	This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable.	

### Example

```
C:\> $getHardwareProfile = Get-UcsHardwareProfile -Credential $cecCred -Name
"Test-HardwareProfile_0"
```

```
C:\> $getHardwareProfile

CreationDate : 2019-06-20T06:37:07+00:00
LastModified : 2019-06-20T06:37:07+00:00
CecId       :
Id          : 9463
Name       : Test-HardwareProfile_0
UcsServer  : Cisco.Ucs.Core.CiscoUcsServer
```

## Remove-UcsHardwareProfile

The Remove-UcsHardwareProfile cmdlet is used to delete the hardware profile from the HCL tool.

### Syntax



```
Remove-UcsHardwareProfile -Credential <PSCredential> -HardwareProfile <CiscoHardwareProfile>
[-Proxy <WebProxy>] [<CommonParameters>]
Remove-UcsHardwareProfile -Credential <PSCredential> -Id <string> [-Proxy <WebProxy>]
[<CommonParameters>]
Remove-UcsHardwareProfile -Credential <PSCredential> -Name <string> [-Proxy <WebProxy>]
[<CommonParameters>]
```

**Detailed Description**

Property	Description	Required
Name	Specifies the name of the Hardware profile.	True
Id	Specifies the name of the Hardware Profile.	True
Proxy	Specifies the web proxy to be used for the communication with HCL api.	False
Credential	Specifies the Cisco.com credential of user.	True
CommonParameters	This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable.	

**Example**

```
C:\> Remove-UcsHardwareProfile -Credential $cecCred -Name "Test-HardwareProfile_0"
Hardware Profile with Name Test-HardwareProfile_0 removed successfully
```

**Invoke-UcsHclUtility**

The Invoke-UcsHclUtility cmdlet is used to get the recommended driver and firmware versions of the adaptors for the hardware profile available in the HCL tool.

**Syntax**

```
Invoke-UcsHclUtility -Credential <PSCredential> -HardwareProfile <CiscoHardwareProfile>
[-Proxy <WebProxy>] [-Tree] [-Advisories] [-AdvisoryType <string>] [<CommonParameters>]
Invoke-UcsHclUtility -Credential <PSCredential> -Name <string> [-Proxy <WebProxy>] [-Tree]
[-Advisories] [-AdvisoryType <string>] [<CommonParameters>]
Invoke-UcsHclUtility -Credential <PSCredential> -Id <string> [-Proxy <WebProxy>] [-Tree]
[-Advisories] [-AdvisoryType <string>] [<CommonParameters>]
```

**Detailed Description**

Property	Description	Required
Name	Specifies the name of the Hardware profile.	True

Property	Description	Required
Advisories	Specifies the switch parameter to show the Advisories published for the components or adaptors that is part of the hardware profile.	False
AdvisoriesType	Specifies the type of advisory to be fetched.	False
Tree	If specified, the cmdlet displays detailed output in a tree view.  <b>Note</b> If you use -Tree, then you cannot use the output of cmdlet for the pipeline	False
HardwareProfile	Specifies the hardware profile object.	True
Proxy	Specifies the web proxy to be used for the communication with HCL api.	False
Credential	Specifies the Cisco.com credential of user.	True
CommonParameters	This cmdlet supports the common parameters: -Verbose, -Debug, -ErrorAction, -ErrorVariable, -OutBuffer, and -OutVariable.	

### Example

```
C:\> $hclResult = Invoke-UcsHclUtility -Credential $cecCred -Name "Test-HardwareProfile_0"

C:\> $hclResult

CompatibilityProfiles Advisories
-----
{810}

C:\> $hclResult.CompatibilityProfiles

Version           : 4.0(1)
DownloadUrl       :
https://software.cisco.com/download/release.html?mdfid=283862063&flowid=25886&softwareid=283655681
InstallationDocUrl :
http://www.cisco.com/c/en/us/support/servers-unified-computing/ucs-manager/products-installation-guides-list.html
DriverIsoDownload :
https://software.cisco.com/download/home/283862063/type/283853158/release/4.0(1)
HardwareTypes     : Cisco.Ucs.Core.HardwareTypes
Component         : {ST1000NM0045, HUSMR7680BDP301, Micron_5100_MTFDDAV240TCB}
OsVersion         : Windows Server 2019
```

```
OsVendor      : Microsoft
OsVersionId   : 810
OsVendorId    : 1
Notes        : {96 : Requires minimum FW version 4.1.2d}
```

### Get-UcsHclAdvisoryTypes

#### Syntax

To view the advisories for the server model and its components, use the **Get-UcsHclAdvisoryTypes** cmdlet.

```
Get-UcsHclAdvisoryTypes -Credential <pscredential>
[-Proxy <WebProxy>][<CommonParameters>]
```

Property	Description	Required
Credential	Specify the Cisco.com user credential	True
Proxy	Specify the web proxy to be used for communicating with HCL API.	False
CommonParameters	This cmdlet supports the common parameters: Verbose, Debug, ErrorAction, ErrorVariable, WarningAction, WarningVariable, OutBuffer, PipelineVariable, and OutVariable.	

#### Example

```
C:\> $advisoryResult = Get-UcsHclAdvisoryTypes -Credential $cecCred

C:\> $advisoryResult
EOL Advisory
Field Notice
```

