Citrix Provisioning Services 7.15 Transition to PowerShell with Objects from PowerShell (Deprecated) Programmer's Guide

Revision 1

August 2017

Table of Contents

Introduction

This document is in the same format as the "PowerShell (Deprecated) Programmer's Guide". It shows for each command in the "PowerShell (Deprecated) McliPSSnapIn.dll" what command to use in the "PowerShell with Objects Citrix.PVS.SnapIn.dll" to accomplish the same task. The values to use in the "PowerShell with Objects Citrix.PVS.SnapIn.dll" are shown in blue and red.

The "PowerShell with Objects Citrix.PVS.SnapIn.dll" follows all of the PowerShell standards and uses objects for all tasks. The "PowerShell (Deprecated) Programmer's Guide" does not follow PowerShell standards and uses strings for input and outputs strings that are parsed to be used.

For documentation of the specific interfaces, refer to the "Citrix Provisioning Services PowerShell with Objects Programmer's Guide" and "Citrix Provisioning Services PowerShell (Deprecated) Programmer's Guide".

New Command Functionality in Citrix Provisioning Services PowerShell with Objects

The commands below do not have a corresponding command in the "Citrix Provisioning Services PowerShell (Deprecated)" are listed below:

Clear-PvsConnection

```
Closes the existing SoapServer connection, and if -Persist is specified the
   connection settings in the registry are removed.

  Optional

   SwitchParameter Persist: If -Persist is specified, clear the
                      connection settings in the registry.

  Optional

   SwitchParameter Confirm: The impact of this operation is "low". If -
                  Confirm is specified, the operation will be
                  confirmed. $ConfirmPreference can be set to "low" to
                  have confirmation without the Confirm parameter.
```

Get-PvsADAccount

```
Return a PvsAdAccount object if the named Device Account in the domain is
   found.

  This required

   string Domain: Domain the account is a member of.

  This required

   string Name: Name of the Device for the account.

  PvsADAccount: If successful, the PvsADAccount object is returned.
```

Get the PvsADAccount in the Domain named theDomain for the Device
named theDevice.

Get-PvsADAccount -Domain theDomain -Name theDevice

Get-PvsDeviceDiskLocatorEnabled

Return true if a Device/DiskLocator is enabled.

One of these required

Guid Guid or DeviceId: Device GUID, to see if the DiskLocator for it
is enabled.

string Name or DeviceName: Device name, to see if the DiskLocator for
it is enabled.

PvsPhysicalAddress[] DeviceMac: MAC of the Device, to see if the
DiskLocator for it is enabled.

This required

Guid DiskLocatorId: DiskLocator GUID, to see if it is enabled for the
Device.

or this required & resolution

string DiskLocatorName: DiskLocator name, to see if it is enabled for
the Device.

One of these resolutions when needed

Guid StoreId: GUID of the Store that is needed when a DiskLocatorName
is used.

string StoreName: Name of the Store that is needed when a
DiskLocatorName is used.

Instead of a parameter that matches one of the members listed

PvsObject[] Object: PvsObjects with the members below can be used as
the Object parameter or from a pileline:

DeviceId or DiskLocatorId

String: If successful, the String value is returned.

Get-PvsDeviceDiskLocatorEnabled -Name theDevice -DiskLocatorName
theDiskLocator -StoreName theStore

DiskLocatorId can be used instead of DiskLocatorName so that the
StoreName or StoreId are not also needed.

Get-PvsConnection

Return the PvsConnection object with the information about the SoapServer
connection.

PvsConnection: If successful, the PvsConnection object is returned.

Get the PvsConnection for the SoapServer.

```
            Get-PvsConnection
```

PvsConnection

```
Read/Write Fields
```

  string Domain: Domain name to use for Authentication. If it has a value, it
    will be *****. Default=""

  string Password: Password to use for Authentication. If it has a value, it
    will be *****. Default=""

  string Persist: True when the connection settings should be, for Set, or
    have been, for Get, saved to the registry.

  string Port: The Port to use to connect. Default=54321

  string Name or Server: Name or IP of the Server to connect to.
    Default=localhost

  string User: User name to use for Authentication. If it has a value, it
    will be *****. Default=""

```
Read-Only Field
```

  string Connected: True when the Citrix.PVS.SnapIn is currently connected to
    the SoapServer with the settings in this PvsConnection.

PvsConnection can be created or modified using methods below:

 New-Object Citrix.PVS.SnapIn.PvsConnection: Creates default
   Server=localhost, Port=54321, and no authentication.

 New-Object Citrix.PVS.SnapIn.PvsConnection(Citrix.PVS.SnapIn copyFrom):
   Creates with settings of the copyFrom Citrix.PVS.SnapIn.

 SetServerToLocalHostDefaultSettings: Server=localhost, Port=54321, and no
   authentication.

 Copy(Citrix.PVS.SnapIn copyFrom): Modifies the settings to match the
   copyFrom Citrix.PVS.SnapIn.

 Equals(Citrix.PVS.SnapIn compareTo): Returns true when the settings match
   what is in the compareTo.

Set-PvsConnection

  Set the SoapServer connection, and if -Persist is specified the connection
    settings are saved in the registry. A PvsConnection object can be used as
    the parameter.

     Required

      PvsConnection Connection: PvsConnection object with changed property
                      value(s) to be set. The object can come from a
                      pileline.

      These values are in the PvsConnection object, and only will be set if
                      the value has changed.

      string Name or Server: Name or IP of the Server to connect to.
                      Default=localhost

      string Port: The Port to use to connect. Default=54321

string User: User name to use for Authentication. If it has a value,
             it will be *****. Default=""

string Domain: Domain name to use for Authentication. If it has a
             value, it will be *****. Default=""

string Password: Password to use for Authentication. If it has a
             value, it will be *****. Default=""

string Persist: True when the connection settings should be, for Set,
             or have been, for Get, saved to the registry.

PvsConnection can be created or modified using methods below:

 New-Object Citrix.PVS.SnapIn.PvsConnection: Creates default
   Server=localhost, Port=54321, and no authentication.

 New-Object Citrix.PVS.SnapIn.PvsConnection(Citrix.PVS.SnapIn
   copyFrom): Creates with settings of the copyFrom Citrix.PVS.SnapIn.

 SetServerToLocalHostDefaultSettings: Server=localhost, Port=54321, and
   no authentication.

 Copy(Citrix.PVS.SnapIn copyFrom): Modifies the settings to match the
   copyFrom Citrix.PVS.SnapIn.

 Equals(Citrix.PVS.SnapIn compareTo): Returns true when the settings
   match what is in the compareTo.

When Connection is not passed, the parameters below are used:

Optional field values to set:

 string Name or Server: Name or IP of the Server to connect to.
             Default=localhost

 string Port: The Port to use to connect. Default=54321

 string User: User name to use for Authentication. If it has a value,
             it will be *****. Default=""

 string Domain: Domain name to use for Authentication. If it has a
             value, it will be *****. Default=""

 string Password: Password to use for Authentication. If it has a
             value, it will be *****. Default=""

 string Persist: True when the connection settings should be, for Set,
             or have been, for Get, saved to the registry.

 Optional

 SwitchParameter PassThru: If -PassThru is specified, the resulting
             PvsConnection object is returned.

 SwitchParameter Confirm: The impact of this operation is "low". If -
             Confirm is specified, the operation will be
             confirmed. $ConfirmPreference can be set to "low" to
             have confirmation without the Confirm parameter.

EXAMPLE 1: Set PvsConnection for Individual Fields

```
Get the PvsConnection into a $o variable. Change the $o field values
                and then Set the PvsConnection with the result.

$o = Get-PvsConnection -Fields Port

$o.Port = 54322

Set-PvsConnection $o

The -Fields parameter with only the needed fields specified makes the
                Get work faster because only those fields are
                retrieved.
```

EXAMPLE 2: Set PvsConnection for a Field Using Pipe

```
Get the PvsConnection into a $o variable for the field that has the
                wrong value. Change the $o field to the correct value
                and then Set the PvsConnection with the result.

Get-PvsConnection -Fields Port | Where-Object {$_.Port -ne 54322} |
                foreach { $o = $_; $o.Port = 54322; $o } | Set-
                PvsConnection

The -Fields parameter with only the needed fields specified makes the
                Get work faster because only those fields are
                retrieved.

The "foreach { $o = $_; $o.X = Y; $o }" sets the field X to value Y
                and returns the object again so it can be piped to
                the Set command for update.
```

EXAMPLE 3: Set PvsConnection Port with Parameter

```
Set the PvsConnection Port using the Port parameter instead of a
                PvsConnection object.

Set-PvsConnection -Port 54322

This is the only Set command that has field parameters.
```

Deprecated Commands Not Supported in Citrix Provisioning Services PowerShell with Objects

The deprecated commands listed below that exist in the "Citrix Provisioning Services PowerShell (Deprecated)" are not included or are replaced with a totally different command that performs the same functionality:

Mcli-Get AdDomains	Get all of the available Active Directory domains. (Get-ADDomain supports same functionality)
Mcli-Get CommandDescription	Return one or more commands with description. (Get-Help *command* supports same functionality)
Mcli-Get CommandFields	Return one record with fields and descriptions. (Get-Help *command* supports same functionality)

Mcli-Get CommandParameters	Return records with parameter type, parameters and descriptions. Parameter types that maybe returned include: Optional, RequiredSingle, OptionalSingle, RequiredSingleAmbiguous, OptionalSingleAmbiguous, and Resolution. (Get-Help *command* supports same functionality)
Mcli-Get CommandSortFields	Return one record with sort fields and descriptions. (Get-Help *command* supports same functionality)
Mcli-Get CommandStandardParameters	Return one record with standard parameter types and descriptions. Parameter types that maybe returned include: parameters, fields, record, records, sortField, resultId, rv, startPosition, maxRecords, totalCount, dataIn, dataOut, and commands. (Get-Help *command* supports same functionality)
Mcli-Get DeviceCustomProperty	Get the Device Custom Property names and values. (Functionality added for previous requirement, but never used)
Mcli-Get DeviceDiskPrinters	Get all non-disabled Printers for a Device/Disk, and the setting for each. (Used for Printer Management for Devices that can be done using Active Directory)
Mcli-Get DiskPrinters	Get all available Printers for a Disk. (Used for Printer Management for Devices that can be done using Active Directory)
Mcli-Get DiskLocatorCustomProperty	Get the DiskLocator Custom Property names and values. (Functionality added for previous requirement, but never used)
Mcli-Get ExceptionDescriptions	Return all of the exceptions with description. (Documented in "Citrix Provisioning Services PowerShell with Objects Programmer's Guide")
Mcli-Get Groups	Get all of the available groups. (Get-ADGroup supports same functionality)
Mcli-Get ServerCustomProperty	Get the Server Custom Property names and values. (Functionality added for previous requirement, but never used)
Mcli-Info MapiErrorCode	Return the string and numeric versions of each Mapi error code. (Documented in "Citrix

	Provisioning Services PowerShell with Objects Programmer's Guide")
Mcli-Run Boot	Deprecated: use RunWithReturn Boot. Boot a Device, Collection or View. (Start-PvsDeviceBoot supports same functionality)
Mcli-Run DisableCollection	Disable a Collection so Devices that exist in it are Disabled, even if a Device itself is set to be Enabled. (Set-Collection supports same functionality)
Mcli-Run DisableDevice	Disable one or more Devices. (Set-Device supports same functionality)
Mcli-Run DisableDiskLocator	Disable a DiskLocator so all Devices that have it assigned will not make it available, even if a Device DiskLocator itself is set to be Enabled. (Set-DiskLocator supports same functionality)
Mcli-Run DisplayMessage	Deprecated: use RunWithReturn DisplayMessage. Display a message on a Device, a Site, Collection or View. (Start-PvsDisplayMessage supports same functionality)
Mcli-Run EnableCollection	Enable a Collection so Devices that exist in it are Enabled, if the Device itself is not set to be Disabled. (Set-Collection supports same functionality)
Mcli-Run EnableDevice	Enable one or more Devices. If the Device's Collection is Disabled, that overrides the Device setting. (Set-Device supports same functionality)
Mcli-Run EnableDiskLocator	Enable a DiskLocator so Devices that have it assigned will make it available, if the Device DiskLocator itself is not set to be Disabled. (Set-DiskLocator supports same functionality)
Mcli-Run ImportDatabase	Import Devices from the version 4.x mdb database fileName specified. (Version 4.x upgrade has been deprecated for many years)
Mcli-Run Reboot	Deprecated: use RunWithReturn Reboot. Reboot a Device, Collection or View. (Start-PvsDeviceReboot supports same functionality)
Mcli-Run SetupConnection	Setup the SOAP server connection that will be used for the MCLI and PowerShell command

	line interfaces. (Set-PvsConnection supports the same functionality)
Mcli-Run Shutdown	Deprecated: use RunWithReturn Shutdown. Shutdown a Device, Collection or View. (Start-PvsDeviceShutdown supports same functionality)
Mcli-Run UnloadConnection	Unload the SOAP server connection so that it can be reconnected when needed for the PowerShell command line interfaces. (Clear-PvsConnection supports the same functionality)
Mcli-RunWithReturn CommandActions	Get a comma delimited list of Command Actions. (Standard PowerShell Verbs are now used instead of Actions)
Mcli-RunWithReturn Enabled	Return 1 if a Collection, Device, DiskLocator, or Device/DiskLocator is enabled. If a collectionName or diskLocatorName is specified, siteName or siteId must be included. If both the Device and DiskLocator are specified, the Enabled setting for the combination is returned. (Get-Collection, Get-Device, Get-DiskLocator and Get-PvsDeviceDiskLocatorEnabled support the same functionality)
Mcli-SetList DeviceCustomProperty	Set the Device Custom Property names and values. (Functionality added for previous requirement, but never used)
Mcli-SetList DeviceDiskPrinters	Set one or more Printer settings. Only the changed items need to be passed. To remove a Printer, set it as disabled. (Used for Printer Management for Devices that can be done using Active Directory)
Mcli-SetList DiskLocatorCustomProperty	Set the DiskLocator Custom Property names and values. (Functionality added for previous requirement, but never used)
Mcli-SetList ServerCustomProperty	Set the Server Custom Property names and values. (Functionality added for previous requirement, but never used)

Short command list

Mcli-Add

Mcli-Add AuthGroup	New-PvsAuthGroup
Mcli-Add CeipData	New-PvsCeipData
Mcli-Add CisData	New-PvsCisData
Mcli-Add Collection	New-PvsCollection
Mcli-Add Device	New-PvsDevice
Mcli-Add DeviceWithPersonalvDisk	New-PvsDeviceWithPersonalvDisk
Mcli-Add DiskLocator	New-PvsDiskLocator
Mcli-Add DiskUpdateDevice	New-PvsDiskUpdateDevice
Mcli-Add FarmView	New-PvsFarmView
Mcli-Add Server	New-PvsServer
Mcli-Add Site	New-PvsSite
Mcli-Add SiteView	New-PvsSiteView
Mcli-Add Store	New-PvsStore
Mcli-Add UpdateTask	New-PvsUpdateTask
Mcli-Add VirtualHostingPool	New-PvsVirtualHostingPool

Mcli-Delete

Mcli-Delete AuthGroup	Remove-PvsAuthGroup
Mcli-Delete Collection	Remove-PvsCollection
Mcli-Delete Device	Remove-PvsDevice
Mcli-Delete DeviceDiskCacheFile	Remove-PvsDeviceDiskCacheFile
Mcli-Delete DiskLocator	Remove-PvsDiskLocator
Mcli-Delete DiskUpdateDevice	Remove-PvsDiskUpdateDevice
Mcli-Delete DiskVersion	Remove-PvsDiskVersion
Mcli-Delete FarmView	Remove-PvsFarmView
Mcli-Delete Server	Remove-PvsServer
Mcli-Delete ServerStore	Remove-PvsServerStore
Mcli-Delete Site	Remove-PvsSite
Mcli-Delete SiteView	Remove-PvsSiteView
Mcli-Delete Store	Remove-PvsStore
Mcli-Delete UpdateTask	Remove-PvsUpdateTask

```
Mcli-Delete VirtualHostingPool                      Remove-
    PvsVirtualHostingPool
```

Mcli-Get

```
Mcli-Get AuditActionParameters                      Get-
    PvsAuditActionParameter

Mcli-Get AuditActionProperties                      Get-
    PvsAuditActionProperty

Mcli-Get AuditActionSibling                         Get-
    PvsAuditActionSibling

Mcli-Get AuditTrail                                 Get-PvsAuditTrail

Mcli-Get AuthGroup                                  Get-PvsAuthGroup

Mcli-Get AuthGroupUsage                             Get-PvsAuthGroupUsage

Mcli-Get CeipData                                   Get-PvsCeipData

Mcli-Get CisData                                    Get-PvsCisData

Mcli-Get Collection                                 Get-PvsCollection

Mcli-Get Device                                     Get-PvsDevice

Mcli-Get DeviceBootstraps                           Get-PvsDeviceBootstrap

Mcli-Get DeviceDiskTempVersion                      Get-
    PvsDeviceDiskTempVersion

Mcli-Get DeviceInfo                                 Get-PvsDeviceInfo

Mcli-Get DevicePersonality                          Get-
    PvsDevicePersonality

Mcli-Get DeviceStatus                               Get-PvsDeviceStatus

Mcli-Get Disk                                       Get-PvsDisk

Mcli-Get DiskInfo                                   Get-PvsDiskInfo

Mcli-Get DiskInventory                              Get-PvsDiskInventory

Mcli-Get DiskLocator                                Get-PvsDiskLocator

Mcli-Get DiskLocatorLock                            Get-PvsDiskLocatorLock

Mcli-Get DiskUpdateDevice                           Get-
    PvsDiskUpdateDevice

Mcli-Get DiskUpdateStatus                           Get-
    PvsDiskUpdateStatus

Mcli-Get DiskVersion                                Get-PvsDiskVersion

Mcli-Get Farm                                       Get-PvsFarm

Mcli-Get FarmView                                   Get-PvsFarmView

Mcli-Get NewVersionDisks                            Get-PvsNewDiskVersion

Mcli-Get Server                                     Get-PvsServer

Mcli-Get ServerBiosBootstrap                        Get-
    PvsServerBiosBootstrap

Mcli-Get ServerBootstrap                            Get-PvsServerBootstrap
```

```
Mcli-Get ServerBootstrapNames                      Get-
    PvsServerBootstrapName

Mcli-Get ServerInfo                                Get-PvsServerInfo

Mcli-Get ServerStatus                              Get-PvsServerStatus

Mcli-Get ServerStore                               Get-PvsServerStore

Mcli-Get Site                                      Get-PvsSite

Mcli-Get SiteView                                  Get-PvsSiteView

Mcli-Get Store                                     Get-PvsStore

Mcli-Get StoreSharedOrServerPath                   Get-
    PvsStoreSharedOrServerPath

Mcli-Get Task                                      Get-PvsTask

Mcli-Get UndefinedDisks                            Get-PvsUndefinedDisk

Mcli-Get UpdateTask                                Get-PvsUpdateTask

Mcli-Get VirtualHostingPool                        Get-
    PvsVirtualHostingPool

Mcli-Get XDSite                                    Get-PvsXDSite
```

Mcli-Info

```
Mcli-Info Group                                    Get-PvsGroup

Mcli-Info LocalServer                              Get-PvsLocalServer

Mcli-Info Version                                  Get-PvsVersion
```

Mcli-Run

```
Mcli-Run ActivateDevice                            Invoke-
    PvsActivateDeviceMAK

Mcli-Run AddDeviceToDomain                         Add-PvsDeviceToDomain

Mcli-Run AddDiskVersion                            Add-PvsDiskVersion

Mcli-Run ApplyAutoUpdate                           Start-PvsAutoUpdate

Mcli-Run ArchiveAuditTrail                         Export-PvsAuditTrail

Mcli-Run AssignAuthGroup                           Grant-PvsAuthGroup

Mcli-Run AssignDevice                              Move-
    PvsDeviceToCollection & Add-PvsDeviceToView

Mcli-Run AssignDisk                                Add-
    PvsDiskToUpdateTask

Mcli-Run AssignDiskLocator                         Add-
    PvsDiskLocatorToDevice

Mcli-Run AssignServer                              Move-PvsServerToSite

Mcli-Run CancelTask                                Stop-PvsTask

Mcli-Run ClearTask                                 Clear-PvsTask

Mcli-Run CopyPasteDevice                           Copy-
    PvsDeviceProperties
```

Mcli-Run CopyPasteDisk	Copy-PvsDiskProperties
Mcli-Run CopyPasteServer	Copy-PvsServerProperties
Mcli-Run CreateDirectory	New-PvsDirectory
Mcli-Run CreateDiskCancel	Stop-PvsCreateDisk
Mcli-Run DisableDeviceDiskLocator	Disable-PvsDeviceDiskLocator
Mcli-Run DiskUpdateCancel	Stop-PvsUpdateTask
Mcli-Run DiskUpdateStart	Start-PvsUpdateTask
Mcli-Run EnableDeviceDiskLocator	Enable-PvsDeviceDiskLocator
Mcli-Run ExportDisk	Export-PvsDisk
Mcli-Run ExportOemLicenses	Export-PvsOemLicenses
Mcli-Run ForceInventory	Update-PvsInventory
Mcli-Run ImportDevices	Import-PvsDevices
Mcli-Run ImportOemLicenses	Import-PvsOemLicenses
Mcli-Run MarkDown	Invoke-PvsMarkDown
Mcli-Run MergeDisk	Merge-PvsDisk
Mcli-Run PromoteDiskVersion	Invoke-PvsPromoteDiskVersion
Mcli-Run RemoveAuthGroup	Revoke-PvsAuthGroup
Mcli-Run RemoveDevice	Remove-PvsDeviceFromView
Mcli-Run RemoveDeviceFromDomain	Remove-PvsDeviceFromDomain
Mcli-Run RemoveDirectory	Remove-PvsDirectory
Mcli-Run RemoveDisk	Remove-PvsDiskFromUpdateTask
Mcli-Run RemoveDiskLocator	Remove-PvsDiskLocatorFromDevice
Mcli-Run ResetDatabaseConnection	Reset-PvsDatabase
Mcli-Run ResetDeviceForDomain	Reset-PvsDeviceForDomain
Mcli-Run RestartStreamService	Restart-PvsStreamService
Mcli-Run RevertDiskVersion	Invoke-PvsRevertDiskVersion
Mcli-Run SetOverrideVersion	Set-PvsOverrideVersion
Mcli-Run StartDeviceDiskTempVersionMode	Start-PvsDeviceDiskTempVersionMode
Mcli-Run StartStreamService	Start-PvsStreamService

```
Mcli-Run StopDeviceDiskTempVersionMode          Stop-
    PvsDeviceDiskTempVersionMode

Mcli-Run StopStreamService                      Stop-PvsStreamService

Mcli-Run UnlockAllDisk                          Unlock-PvsAllDisk

Mcli-Run UnlockDisk                             Unlock-PvsDisk

Mcli-Run UnmapDisk                              Dismount-PvsDisk

Mcli-Run ValidateDirectory                      Test-PvsDirectory
```

Mcli-RunWithReturn

```
Mcli-RunWithReturn Boot                         Start-PvsDeviceBoot

Mcli-RunWithReturn CreateDisk                   Start-PvsCreateDisk

Mcli-RunWithReturn CreateDiskStatus             Get-
    PvsCreateDiskStatus

Mcli-RunWithReturn CreateMaintenanceVersion     New-
    PvsDiskMaintenanceVersion

Mcli-RunWithReturn DeviceCount                  Get-PvsDeviceCount

Mcli-RunWithReturn DiskLocatorCount             Get-
    PvsDiskLocatorCount

Mcli-RunWithReturn DisplayMessage               Start-
    PvsDisplayMessage

Mcli-RunWithReturn Exists                       Get-PvsExists

Mcli-RunWithReturn ImportDisk                   Import-PvsDisk

Mcli-RunWithReturn ListDirectories              Get-PvsDirectory

Mcli-RunWithReturn MaintenanceVersionExists     Get-
    PvsMaintenanceVersionExists

Mcli-RunWithReturn MapDisk                      Mount-PvsDisk

Mcli-RunWithReturn MappedDiskId                 Get-PvsMountedDisk

Mcli-RunWithReturn MappedDriveLetter            Get-
    PvsMountedDriveLetter

Mcli-RunWithReturn MinimumLastAutoAddDeviceNumber  Get-
    PvsMinimumLastAutoAddDeviceNumber

Mcli-RunWithReturn RebalanceDevices             Invoke-
    PvsRebalanceDevices

Mcli-RunWithReturn Reboot                       Start-PvsDeviceReboot

Mcli-RunWithReturn ReportBug                    Start-PvsReportBug

Mcli-RunWithReturn ServerCount                  Get-PvsServerCount

Mcli-RunWithReturn ServerName                   Get-PvsServerName

Mcli-RunWithReturn ServerStoreActiveDeviceCount  Get-
    PvsServerStoreActiveDeviceCount

Mcli-RunWithReturn Shutdown                     Start-
    PvsDeviceShutdown
```

```
Mcli-RunWithReturn StoreFreeSpace                    Get-PvsStoreFreeSpace

Mcli-RunWithReturn TaskStatus                        Get-PvsTaskStatus

Mcli-RunWithReturn UpdateBDM                         Start-
   PvsDeviceUpdateBdm

Mcli-RunWithReturn UploadCeip                        Get-PvsUploadCeip
```

Mcli-Set

```
Mcli-Set AuthGroup                                   Set-PvsAuthGroup

Mcli-Set CeipData                                    Set-PvsCeipData

Mcli-Set CisData                                     Set-PvsCisData

Mcli-Set Collection                                  Set-PvsCollection

Mcli-Set Device                                      Set-PvsDevice

Mcli-Set Disk                                        Set-PvsDisk

Mcli-Set DiskLocator                                 Set-PvsDiskLocator

Mcli-Set DiskUpdateDevice                            Set-
   PvsDiskUpdateDevice

Mcli-Set DiskVersion                                 Set-PvsDiskVersion

Mcli-Set Farm                                        Set-PvsFarm

Mcli-Set FarmView                                    Set-PvsFarmView

Mcli-Set Server                                      Set-PvsServer

Mcli-Set ServerBiosBootstrap                         Set-
   PvsServerBiosBootstrap

Mcli-Set ServerBootstrap                             Set-PvsServerBootstrap

Mcli-Set ServerStore                                 Set-PvsServerStore

Mcli-Set Site                                        Set-PvsSite

Mcli-Set SiteView                                    Set-PvsSiteView

Mcli-Set Store                                       Set-PvsStore

Mcli-Set UpdateTask                                  Set-PvsUpdateTask

Mcli-Set VirtualHostingPool                          Set-
   PvsVirtualHostingPool

Mcli-Set XDSite                                      Set-PvsXDSite
```

Mcli-SetList

```
Mcli-SetList DeviceBootstraps                        Set-PvsDeviceBootstrap

Mcli-SetList DevicePersonality                       Set-
   PvsDevicePersonality
```

Commands by Object Type

Many

Mcli-Get AuditActionParameters	Get-PvsAuditActionParameter
Mcli-Get AuditActionProperties	Get-PvsAuditActionProperty
Mcli-Get AuditActionSibling	Get-PvsAuditActionSibling
Mcli-Get AuditTrail	Get-PvsAuditTrail
Mcli-Run ResetDatabaseConnection	Reset-PvsDatabase
Mcli-RunWithReturn Exists	Get-PvsExists

AuthGroup

Mcli-Add AuthGroup	New-PvsAuthGroup
Mcli-Delete AuthGroup	Remove-PvsAuthGroup
Mcli-Get AuthGroup	Get-PvsAuthGroup
Mcli-Get AuthGroupUsage	Get-PvsAuthGroupUsage
Mcli-Run AssignAuthGroup	Grant-PvsAuthGroup
Mcli-Run RemoveAuthGroup	Revoke-PvsAuthGroup
Mcli-Set AuthGroup	Set-PvsAuthGroup

CeipData

Mcli-Add CeipData	New-PvsCeipData
Mcli-Get CeipData	Get-PvsCeipData
Mcli-RunWithReturn UploadCeip	Get-PvsUploadCeip
Mcli-Set CeipData	Set-PvsCeipData

CisData

Mcli-Add CisData	New-PvsCisData
Mcli-Get CisData	Get-PvsCisData
Mcli-Set CisData	Set-PvsCisData

Collection

Mcli-Add Collection	New-PvsCollection
Mcli-Delete Collection	Remove-PvsCollection
Mcli-Get Collection	Get-PvsCollection
Mcli-RunWithReturn MinimumLastAutoAddDeviceNumber	Get-PvsMinimumLastAutoAddDeviceNumber
Mcli-Set Collection	Set-PvsCollection

Device

Mcli-Add Device	New-PvsDevice
Mcli-Add DeviceWithPersonalvDisk	New-PvsDeviceWithPersonalvDisk
Mcli-Delete Device	Remove-PvsDevice
Mcli-Delete DeviceDiskCacheFile	Remove-PvsDeviceDiskCacheFile
Mcli-Get Device	Get-PvsDevice
Mcli-Get DeviceBootstraps	Get-PvsDeviceBootstrap
Mcli-Get DeviceDiskTempVersion	Get-PvsDeviceDiskTempVersion
Mcli-Get DeviceInfo	Get-PvsDeviceInfo
Mcli-Get DevicePersonality	Get-PvsDevicePersonality
Mcli-Get DeviceStatus	Get-PvsDeviceStatus
Mcli-Run ActivateDevice	Invoke-PvsActivateDeviceMAK
Mcli-Run AddDeviceToDomain	Add-PvsDeviceToDomain
Mcli-Run AssignDevice	Move-PvsDeviceToCollection & Add-PvsDeviceToView
Mcli-Run CopyPasteDevice	Copy-PvsDeviceProperties
Mcli-Run DisableDeviceDiskLocator	Disable-PvsDeviceDiskLocator
Mcli-Run EnableDeviceDiskLocator	Enable-PvsDeviceDiskLocator
Mcli-Run ExportOemLicenses	Export-PvsOemLicenses
Mcli-Run ImportDevices	Import-PvsDevices
Mcli-Run ImportOemLicenses	Import-PvsOemLicenses
Mcli-Run MarkDown	Invoke-PvsMarkDown
Mcli-Run RemoveDevice	Remove-PvsDeviceFromView
Mcli-Run RemoveDeviceFromDomain	Remove-PvsDeviceFromDomain
Mcli-Run ResetDeviceForDomain	Reset-PvsDeviceForDomain
Mcli-Run StartDeviceDiskTempVersionMode	Start-PvsDeviceDiskTempVersionMode
Mcli-Run StopDeviceDiskTempVersionMode	Stop-PvsDeviceDiskTempVersionMode
Mcli-RunWithReturn Boot	Start-PvsDeviceBoot
Mcli-RunWithReturn DeviceCount	Get-PvsDeviceCount

```
Mcli-RunWithReturn DisplayMessage                    Start-
    PvsDisplayMessage

Mcli-RunWithReturn Reboot                            Start-PvsDeviceReboot

Mcli-RunWithReturn Shutdown                          Start-
    PvsDeviceShutdown

Mcli-RunWithReturn UpdateBDM                         Start-
    PvsDeviceUpdateBdm

Mcli-Set Device                                      Set-PvsDevice

Mcli-SetList DeviceBootstraps                        Set-PvsDeviceBootstrap

Mcli-SetList DevicePersonality                       Set-
    PvsDevicePersonality
```

Disk

```
Mcli-Add DiskUpdateDevice                            New-
    PvsDiskUpdateDevice

Mcli-Add VirtualHostingPool                          New-
    PvsVirtualHostingPool

Mcli-Delete DiskUpdateDevice                         Remove-
    PvsDiskUpdateDevice

Mcli-Delete DiskVersion                              Remove-PvsDiskVersion

Mcli-Delete VirtualHostingPool                       Remove-
    PvsVirtualHostingPool

Mcli-Get Disk                                        Get-PvsDisk

Mcli-Get DiskInfo                                    Get-PvsDiskInfo

Mcli-Get DiskInventory                               Get-PvsDiskInventory

Mcli-Get DiskUpdateDevice                            Get-
    PvsDiskUpdateDevice

Mcli-Get DiskUpdateStatus                            Get-
    PvsDiskUpdateStatus

Mcli-Get DiskVersion                                 Get-PvsDiskVersion

Mcli-Get NewVersionDisks                             Get-PvsNewDiskVersion

Mcli-Get UndefinedDisks                              Get-PvsUndefinedDisk

Mcli-Get VirtualHostingPool                          Get-
    PvsVirtualHostingPool

Mcli-Run AddDiskVersion                              Add-PvsDiskVersion

Mcli-Run AssignDisk                                  Add-
    PvsDiskToUpdateTask

Mcli-Run CopyPasteDisk                               Copy-PvsDiskProperties

Mcli-Run CreateDiskCancel                            Stop-PvsCreateDisk

Mcli-Run ExportDisk                                  Export-PvsDisk

Mcli-Run ForceInventory                              Update-PvsInventory

Mcli-Run MergeDisk                                   Merge-PvsDisk
```

```
Mcli-Run PromoteDiskVersion                    Invoke-
   PvsPromoteDiskVersion

Mcli-Run RemoveDisk                            Remove-
   PvsDiskFromUpdateTask

Mcli-Run RevertDiskVersion                     Invoke-
   PvsRevertDiskVersion

Mcli-Run SetOverrideVersion                    Set-PvsOverrideVersion

Mcli-Run UnlockAllDisk                         Unlock-PvsAllDisk

Mcli-Run UnlockDisk                            Unlock-PvsDisk

Mcli-Run UnmapDisk                             Dismount-PvsDisk

Mcli-RunWithReturn CreateDisk                  Start-PvsCreateDisk

Mcli-RunWithReturn CreateDiskStatus            Get-
   PvsCreateDiskStatus

Mcli-RunWithReturn CreateMaintenanceVersion    New-
   PvsDiskMaintenanceVersion

Mcli-RunWithReturn ImportDisk                  Import-PvsDisk

Mcli-RunWithReturn MaintenanceVersionExists    Get-
   PvsMaintenanceVersionExists

Mcli-RunWithReturn MapDisk                     Mount-PvsDisk

Mcli-RunWithReturn MappedDiskId                Get-PvsMountedDisk

Mcli-RunWithReturn MappedDriveLetter           Get-
   PvsMountedDriveLetter

Mcli-Set Disk                                  Set-PvsDisk

Mcli-Set DiskUpdateDevice                      Set-
   PvsDiskUpdateDevice

Mcli-Set DiskVersion                           Set-PvsDiskVersion

Mcli-Set VirtualHostingPool                    Set-
   PvsVirtualHostingPool
```

DiskLocator

```
Mcli-Add DiskLocator                           New-PvsDiskLocator

Mcli-Delete DiskLocator                        Remove-PvsDiskLocator

Mcli-Get DiskLocator                           Get-PvsDiskLocator

Mcli-Get DiskLocatorLock                       Get-PvsDiskLocatorLock

Mcli-Run AssignDiskLocator                     Add-
   PvsDiskLocatorToDevice

Mcli-Run RemoveDiskLocator                     Remove-
   PvsDiskLocatorFromDevice

Mcli-RunWithReturn DiskLocatorCount            Get-
   PvsDiskLocatorCount

Mcli-Set DiskLocator                           Set-PvsDiskLocator
```

Farm

Mcli-Get Farm	Get-PvsFarm
Mcli-Get XDSite	Get-PvsXDSite
Mcli-Run ArchiveAuditTrail	Export-PvsAuditTrail
Mcli-Set Farm	Set-PvsFarm
Mcli-Set XDSite	Set-PvsXDSite

FarmView

Mcli-Add FarmView	New-PvsFarmView
Mcli-Delete FarmView	Remove-PvsFarmView
Mcli-Get FarmView	Get-PvsFarmView
Mcli-Set FarmView	Set-PvsFarmView

Server

Mcli-Add Server	New-PvsServer
Mcli-Delete Server	Remove-PvsServer
Mcli-Delete ServerStore	Remove-PvsServerStore
Mcli-Get Server	Get-PvsServer
Mcli-Get ServerBiosBootstrap	Get-PvsServerBiosBootstrap
Mcli-Get ServerBootstrap	Get-PvsServerBootstrap
Mcli-Get ServerBootstrapNames	Get-PvsServerBootstrapName
Mcli-Get ServerInfo	Get-PvsServerInfo
Mcli-Get ServerStatus	Get-PvsServerStatus
Mcli-Get ServerStore	Get-PvsServerStore
Mcli-Run ApplyAutoUpdate	Start-PvsAutoUpdate
Mcli-Run AssignServer	Move-PvsServerToSite
Mcli-Run CopyPasteServer	Copy-PvsServerProperties
Mcli-Run CreateDirectory	New-PvsDirectory
Mcli-Run RemoveDirectory	Remove-PvsDirectory
Mcli-Run RestartStreamService	Restart-PvsStreamService
Mcli-Run StartStreamService	Start-PvsStreamService
Mcli-Run StopStreamService	Stop-PvsStreamService
Mcli-Run ValidateDirectory	Test-PvsDirectory
Mcli-RunWithReturn ListDirectories	Get-PvsDirectory

```
Mcli-RunWithReturn RebalanceDevices            Invoke-
   PvsRebalanceDevices

Mcli-RunWithReturn ReportBug                   Start-PvsReportBug

Mcli-RunWithReturn ServerCount                 Get-PvsServerCount

Mcli-RunWithReturn ServerName                  Get-PvsServerName

Mcli-RunWithReturn ServerStoreActiveDeviceCount Get-
   PvsServerStoreActiveDeviceCount

Mcli-Set Server                                Set-PvsServer

Mcli-Set ServerBiosBootstrap                   Set-
   PvsServerBiosBootstrap

Mcli-Set ServerBootstrap                       Set-PvsServerBootstrap

Mcli-Set ServerStore                           Set-PvsServerStore
```

Site

```
Mcli-Add Site                                  New-PvsSite

Mcli-Delete Site                               Remove-PvsSite

Mcli-Get Site                                  Get-PvsSite

Mcli-Get StoreSharedOrServerPath               Get-
   PvsStoreSharedOrServerPath

Mcli-Set Site                                  Set-PvsSite
```

SiteView

```
Mcli-Add SiteView                              New-PvsSiteView

Mcli-Delete SiteView                           Remove-PvsSiteView

Mcli-Get SiteView                              Get-PvsSiteView

Mcli-Set SiteView                              Set-PvsSiteView
```

Store

```
Mcli-Add Store                                 New-PvsStore

Mcli-Delete Store                              Remove-PvsStore

Mcli-Get Store                                 Get-PvsStore

Mcli-RunWithReturn StoreFreeSpace              Get-PvsStoreFreeSpace

Mcli-Set Store                                 Set-PvsStore
```

System

```
Mcli-Info Group                                Get-PvsGroup

Mcli-Info LocalServer                          Get-PvsLocalServer

Mcli-Info Version                              Get-PvsVersion
```

Task

```
Mcli-Get Task                                  Get-PvsTask
```

```
Mcli-Run CancelTask                              Stop-PvsTask

Mcli-Run ClearTask                               Clear-PvsTask

Mcli-RunWithReturn TaskStatus                    Get-PvsTaskStatus
```

UpdateTask

```
Mcli-Add UpdateTask                              New-PvsUpdateTask

Mcli-Delete UpdateTask                           Remove-PvsUpdateTask

Mcli-Get UpdateTask                              Get-PvsUpdateTask

Mcli-Run DiskUpdateCancel                        Stop-PvsUpdateTask

Mcli-Run DiskUpdateStart                         Start-PvsUpdateTask

Mcli-Set UpdateTask                              Set-PvsUpdateTask
```

Error codes

For the Citrix.PVS.SnapIn, if an error occurs, a PvsException will be in the Exception member of the $error.

The members of a PvsException are:

InnerException: The exception that occured. This exception maybe an EAException or other standard Exception.

ToString(): Has the formatted full Message of the InnerException.

If the InnerException GetType().Name equals "EAException", then The members of it are:

returnCode: The number, as shown below in the Error codes. The name of the error, for example "NotImplemented", is not included in the EAException.

Message: The message, as shown below in the Error codes. The [v1], [v2], [v3], [v4], and [v5] will be replaced with values as required.

Details: Has the Details for the EAException if there are any. OtherException, ManagementInterfaceError and PvsStatusException will have Details.

ToString(): Has the Message as shown below in the Error codes. If there is Details, it will be returned or included, and if partialReturn, they will be included.

partialReturn: Might have a list of EAException objects if any of the items processed during the command had any issues.

Severity: Can have the values Critical, Error, Warning or Duplicate.

Source: Has the value that is displayed in the Console as a Title or Type for the error.

0 Success: The command succeeded.

1 NotImplemented: The [v1] feature has not been implemented.

2 InvalidCommand: The [v1] command does not exist.

3 InvalidField: The [v1] field does not exist.
```

4 InvalidFieldFormat: The [v1] field is not formatted properly, the correct format is [v2].

5 InvalidParameter: The [v1] parameter is not valid.

6 InvalidParameterFormat: The [v1] parameter is not formatted property, the correct format is [v2].

7 ReadOnlyField: Unable to change the [v1] field because it is read-only.

8 RequiredFieldMissing: The required [v1] field is missing.

9 RequiredFieldsMissing: The required [v1] or [v2] field is missing.

10 RequiredParameterMissing: The required [v1] parameter is missing.

11 RequiredParametersMissing: The required [v1] or [v2] parameter is missing.

12 InternalIdAndNameFieldsMustBeDefined: An internal error occurred. The [v1] field is not the next FieldSettings object after the ID.

13 NoFarmAccess: The domain/user does not have access to the Farm.

14 InvalidForeignKeyValue: The [v1] field with value [v2] is an invalid foreign key.

15 SetupError: The system was not configured correctly.

16 Executing: The [v1] command can only be called one at a time. Wait for the command to finish before running again.

17 NoDiskMapped: A vDisk has not yet been mapped.

18 DatabaseError: A database error occurred.

19 DuplicateKey: To avoid creating a duplicate key, the Add or Set command was cancelled.

20 DatabaseErrorMissed: An internal error occurred. An uncaught database error occurred.

21 AddCommandFailed: No objects were added during the last 'Add' command.

22 InsufficientPrivileges: Access denied. The appropriate privileges are not assigned to perform this task.

23 ZeroObjectsAffected: No object was added, updated, or deleted in the last operation.

24 OtherException: An unexpected MAPI error occurred.

25 InvalidFieldLength: The [v1] field value is too long, maximum length is [v2].

26 InvalidFieldValueMinMax: The [v1] field value is invalid, the minimum is [v2] and the maximum is [v3].

27 InvalidFieldValue: The [v1] field can only have values [v2] or [v3].

28 TooManyParameters: More parameters were specified than permitted.

29 TooFewParameters: Not enough identifying parameters specified.

30 FollowingParametersMissing: To use the [v1] parameter, [v2] or [v3] must also be used.

31 InconsistentData: The action is canceled because the Store directory date/times does not match. Update the Store directories to match.

32 DatabaseOpenFailed: Unable to contact the database server. Ensure Provisioning Services is configured correctly.

33 DatabaseVersionWrong: The wrong database version is being used.  Found version number: [v1] Expected version number: [v2]

34 DatabaseVersionNotFound: The database version number does not exist or was not found. Ensure Provisioning Services is configured correctly.

35 SomeRequiredParametersMissing: Required parameters are missing.

36 PartialError: The following items failed:

   item1 Error message...

   item2 Error message...

37 NoItemsToProcess: There are no items to process.

38 NoDefaultCollectionDefined: Unable to add a Device until a default Collection is set.

39 NoDefaultSiteDefined: A default Site is not set, no Devices can be added.

40 InvalidCollection: The specified Collection does not exist.

41 InvalidAuthGroup: The specified AuthGroup does not exist.

42 InvalidGroup: The specified Group does not exist.

43 InvalidDevice: The specified Device does not exist.

44 InvalidDiskLocator: The specified vDisk does not exist.

45 InvalidServer: The specified Server does not exist.

46 InvalidServerSite: Server specified is not in the Site specified.

47 InvalidStoreSite: Store specified is not for the Site specified.

48 InvalidSiteView: The specified Site View does not exist.

49 InvalidSite: The specified Site does not exist.

50 InvalidDeviceDiskLocator: The specified Device or vDisk does not exist.

51 InvalidDeviceImport: Import failed because the file must have Device Name, Mac Address, Site Name, and Collection Name, and they must be tab or comma-delimited.

52 InvalidServerFrom: The Server to copy [v1]=[v2] was not found.

53 InvalidServerTo: No Server to copy to ([v1]=[v2]) was found.

54 InvalidDeviceFrom: The Device to copy [v1]=[v2] was not found.

55 InvalidDeviceTo: No Devices to copy to are found.

56 InvalidDiskFrom: The vDisk to copy [v1]=[v2] was not found.

57 InvalidDiskTo: No vDisk to copy to ([v1]=[v2]) was found.

58 InvalidDiskPath: The path '[v1]' to the vDisk file is not found.

59 VDiskFileNotFound: [v1]: vDisk file was not found.

60 InvalidDiskServer: There is no Server that can serve the vDisk [v1] or the Store to which this vDisk belongs. Verify that one or more Servers belonging to the Store are online and that there is sufficient free space for the operation you are attempting.

61 InvalidDiskForServer: Server [v1] cannot access all versions of vDisk [v2], the vDisk was updated on at least one other Server.

62 SameSiteRequired: Objects within the same Site must be selected.

63 TooFewFields: Not enough fields for a record.

64 ADerrorDC: Unable to connect to the Domain Controller (if any) or the default rootDSE. Error code: [v1], message: [v2], provider: [v3].

65 ADerrorOU: Unable to get the Organizational Unit setting (if any). Error code: [v1], message: [v2], provider: [v3].

66 ADerrorDefaultContainer: Unable to get the default computer accounts container (default location is Active Directory root> Computers). Error code: [v1], message: [v2], provider: [v3].

67 ADerrorCreate: Unable to create the computer account in Active Directory. Ensure the account does not already exist and that the appropriate permissions are available to perform this task. Error code: [v1], message: [v2], provider: [v3].

68 ADerrorNewAccount: Unable to get the newly created Active Directory computer account. Error code: [v1], message: [v2], provider: [v3].

69 ADerrorSam: Unable to set the Active Directory samAccountName property. Ensure the appropriate permissions exist to perform this task. Error code: [v1], message: [v2], provider: [v3].

70 ADerrorUserAccount: Unable to set the Active Directory userAccountControl property. Ensure the appropriate permissions exist to perform this task. Error code: [v1], message: [v2], provider: [v3].

71 ADerrorSave: Unable to save Active Directory change. Ensure the appropriate permissions exist to perform this task. Error code: [v1], message: [v2], provider: [v3].

72 ADerrorSetPassword: Unable to set a new password for this user account. Ensure the appropriate permissions exist to perform this task. Error code: [v1], message: [v2], provider: [v3].

73 ADerrorAddTrustee: Unable to add trustee (if any). Error code: [v1], message: [v2], provider: [v3].

74 ADerrorEnableAccount: Unable to enable the Active Directory account. Error code: [v1], message: [v3], provider: [v2].

75 ADerrorAlreadyExists: The computer name is already in use. Error code: [v1], message: [v3], provider: [v2]. Select a unique name for this machine.

76 ADerrorGeneral: A general Active Directory error occurred. Error code: [v1], message: [v2], provider: [v3].

77 ADerrorDirectorySearch: Unable to find Active Directory items meeting the search criteria entered. Error code: [v1], message: [v2], provider: [v3].

78 ADerrorSearchComputerAccount: Unable to perform the computer accounts search. Error code: [v1], message: [v2], provider: [v3].

79 ADerrorComputerAccountNotFound: Specified computer account not found. Error code: [v1], message: [v2], provider: [v3].

80 ADerrorComputerAccountHold: This computer account is currently unavailable. Ensure that Active Directory is running properly. Error code: [v1], message: [v2], provider: [v3].

81 ADerrorComputerAccountMove: Failed to move the computer account to the target organizational unit set (also returned if caller lacks permission). Error code: [v1], message: [v2], provider: [v3].

82 ADerrorDelete: Unable to delete this computer account. Ensure the appropriate permissions exist to perform this task. Error code: [v1], message: [v2], provider: [v3].

83 ADerrorPasswordGeneration: Unable to generate this password. Ensure the appropriate permissions exist to perform this task.

84 MapDiskNoDriver: Unable to map vDisk because a driver was not found.

85 MapDiskDeniedByServer: Unable to map the vDisk. Mapping was denied by the Server.

86 MapDiskLocalAccessDenied: Unable to map the vDisk. Denied local access.

87 MapDiskMiniportError: Unable to map vDisk because of a Miniport error.

88 UnmapDiskFailed: Failed to unmap a vDisk.

89 DuplicateDisk: The vDisk [v1] already exists on [v2] at [v3].

90 DuplicateDiskLocator: A DiskLocator: [v1] already exists on Site: [v2].

91 DiskCreationInProgress: The vDisk [v1] is being created on [v2] at [v3].

92 InvalidServerStore: A database integrity error occurred. The Server is not set to deliver vDisks from the Store, but should be.

93 InvalidStore: The specified Store does not exist.

94 InvalidFarmView: Farm View specified does not exist.

95 InvalidStorePath: Store path is empty.

96 ManagementInterfaceError:

Management Interface: Undefined error.

Management Interface: Database interface is inaccessible.

Management Interface: Database interface library is inaccessible.

Management Interface: The database access library is a version incompatible with the Management Server.

Management Interface: Database interface library is invalid.

Management Interface: Database interface could not be created.

Management Interface: Database could not be opened.

Management Interface: Database is in use.

Management Interface: Database error occurred.

Management Interface: Not implemented.

Management Interface: Registry entry was not found.

Management Interface: Request was not created.

Management Interface: Operating System error occurred.

Management Interface: vDisk error.

Management Interface: vDisk header is incomplete.

Management Interface: vDisk footer is incomplete.

Management Interface: vDisk boot record is incomplete.

Management Interface: vDisk boot sector is incomplete.

Management Interface: vDisk size is below the minimum.

Management Interface: vDisk size is above the maximum.

Management Interface: vDisk boot record template is inaccessible.

Management Interface: vDisk boot sector template is inaccessible.

Management Interface: vDisk lock was not found.

Management Interface: vDisk has exclusive lock.

Management Interface: vDisk has shared lock.

Management Interface: vDisk lock error.

Management Interface: vDisk format is incompatible.

Management Interface: vDisk prefooter is incomplete.

Management Interface: vDisk creation is in progress.

Management Interface: vDisk creation information was not found.

Management Interface: vDisk creation cancellation was requested.

Management Interface: vDisk file was not found.

Management Interface: vDisk file path was not found.

Management Interface: vDisk file access was denied.

Management Interface: Cancelled.

Management Interface: Registry key for the product is inaccessible.

Management Interface: Registry key for the installation folder is
    inaccessible.

Management Interface: Registry key for the management interface is
    inaccessible.

Management Interface: Registry key for the database path is
    inaccessible.

Management Interface: Registry key for the management interface IP
    address is inaccessible.

Management Interface: Buffer size is too small.

Management Interface: Buffer size is too large.

Management Interface: Unknown error.

Management Interface: Remote Server failed to relay a request.

Management Interface: Remote Server is not servicing the Device.

Management Interface: Remote Server or Device refused the request.

Management Interface: Local Server failed to complete a request to a
    Server or Device.

Management Interface: Local Server failed to complete a request to a Server.

Management Interface: Remote requests were disabled because of an initialization error.

Management Interface: Remote request failed.

Management Interface: Remote request timed out.

Management Interface: Remote request result was not found.

Management Interface: Remote request receiver failed to initialize.

Management Interface: Management command failed for all objects.

Management Interface: Failed to get the preshared key in secure version.

Management Interface: VHD Error.

Management Interface: vDisk properties were lost.

Management Interface: Insufficient Memory.

Management Interface: The network path was not found.

Management Interface: The network name cannot be found.

Management Interface: File already exists.

Management Interface: The geometry of the vDisk is not accessible.

Management Interface: Unable to create the vDisk because the store media is read-only.

Management Interface: vDisk file is being used by another process.

97 ServerTimeout: Server did not respond to a request in time.

98 NotFound: [v1] not found.

99 AccountRetrieve: Account information for user [v1] was not found.

100 ActiveDevice: The task cannot be performed on active Devices. Shut down the Devices before attempting to perform the task.

101 ActiveDiskLocator: The task cannot be performed on active vDisks. Shut down the Devices that are using the vDisks before attempting to perform the task.

102 AssignedDiskLocator: Unable to delete a vDisk that is currently assigned to a Device. Unassign all Devices, then delete the vDisk.

103 ActiveServer: The task cannot be performed on active Servers. Shut down the Servers before attempting to perform the task.

104 NotEnoughFreeDiskSpace: There is not enough free disk space to create the vDisk.

105 InvalidDiskName: The vDisk name has one or more invalid characters. The invalid characters are < > | " \ / : * ?.

106 CannotDeleteLastAuthGroup: Deleting the last Authorization Group causes the system to be inoperable.

107 CannotDeleteUsedAuthGroup: An Authorization Group that is currently in use cannot be deleted.

108 ServerStartFailed: The Server did not start successfully. Ensure the
    appropriate permissions exist for the service account.

109 ServerStopFailed: The Server did not stop successfully.

110 LockOwnerNotFound: The Device that owns the lock was not found, the
    vDisk was not unlocked.

111 PossiblySharedVDisk: Unable to delete File [v1]. It is possible that the
    file is being referenced in other Sites or Stores.

112 StorePathInaccessible: The Store path [v1] is inaccessible.

113 InvalidAction: The [v1] action does not exist.

114 InvalidObjectType: The [v1] objectType does not exist.

115 TooManyRecords: The amount of data returned using Get is too large. Use
    GetFirst and GetNext instead of Get.

116 InvalidUserGroup: The specified UserGroup does not exist.

117 InvalidAuditAction: The specified AuditAction does not exist.

118 LoginFailed: The database login failed. Ensure the appropriate
    permissions exist to access the database.

119 DatabaseConnectionError: Unable to connect to the database. Restore the
    connection in order to manage the farm.

120 CreateTriggersParsing: Unable to parse the database script
    'CreateTriggers' at: [v1]

121 CreateStoredProcParsing: Unable to parse the database script
    'CreateStoredProcedures' at: [v1]

122 MediaIsReadOnly: Management Interface: Unable to create the vDisk
    because the store media is read-only.

123 ConnectedDeviceForVirtualHostingPool: Unable to delete this VM from a
    machine catalog because it is connected to a Delivery Group.

124 ADerrorDN: Unable to get the distinguishedName property. Ensure the
    appropriate Active Directory permissions exist to perform this task.
    Error code: [v1], message: [v2], provider: [v3].

125 ADerrorGetSecDes: Unable to get the Active Directory Security Descriptor
    property. Error code: [v1], message: [v2], provider: [v3].

126 ADerrorSetSecDes: Unable to set the Active Directory Security Descriptor
    property. Ensure the appropriate permissions exist to perform this
    task. Error code: [v1], message: [v2], provider: [v3].

127 ADerrorDNSHostName: Unable to set the DNS Host Name property
    (dNSHostName). Ensure the appropriate permissions exist to perform
    this task. Error code: [v1], message: [v2], provider: [v3].

128 ADerrorDisplayName: Unable to set the displayName property. Error code:
    [v1], message: [v2], provider: [v3].

129 ADerrorBind: This device was unable to bind to the Domain Controller.
    Ensure the Domain Controller is running. Error code: [v1], message:
    [v2], provider: [v3].

130 ADerrorGetSPN: Unable to get an Active Directory Service Principal Name.
    Error code: [v1], message: [v2], provider: [v3].

131 ADerrorWriteSPN: Unable to write the Active Directory Service Principal Name. Error code: [v1], message: [v2], provider: [v3]

132 ADerrorSearch: Unable to perform the requested Search. Error code: [v1], message: [v2], provider: [v3].

133 ADerrorMoveToOU: Unable to move the Active Directory account to the requested Organizational Unit. Ensure the appropriate permissions exist to perform this task. Error code: [v1], message: [v2], provider: [v3].

134 ADerrorDeleteAccount: Unable to delete this computer account. Ensure the appropriate permissions exist to delete accounts.  Error code: [v1], message: [v2], provider: [v3].

135 ADerrorBadParameters: Incorrect parameters sent to Provisioning Services from  Studio. Error code: [v1], message: [v2], provider: [v3].

136 VolumeInUse: The volume is being used.

137 VolumeAccessDenied: Volume access is denied.

138 VolumeUnknownVolume: An unknown volume was specified.

139 VolumeGeneralError: An error occured when executing a volume command.

140 MaintenanceServerError: Action cannot be performed, [v1] is a maintenance server for [v2].

141 NotManagedStore: The action cannot be performed because the store is not managed.

142 PathNotExist: The path does not exist on the given Server.

143 PathNoCreatePermission: The path does not have the appropriate create permissions.

144 PathNoReadPermission: The path does not have the appropriate read permissions.

145 PathNoWritePermission: The path does not have the appropriate write permissions.

146 PathNoDeletePermission: The path does not have the appropriate delete permissions.

147 IPCProtocolError: An internal error occurred. A field is missing from the process communication protocol data.

148 InvalidStoreServer: No active Server can serve the Store [v1].

149 ConstraintCheck: A database constraint caused an Add or Update to be stopped.

150 VamtNotFound: The Volume Activation Management Tool cannot be found.

151 ADerrorCannotGetObjectSID: Cannot return objectSID. Error code: [v1], message: [v2], provider: [v3].

152 ADerrorCannotDisableAccount: Cannot disable the Active Directory account at this time. Ensure that all account users are logged off before attempting to disable the account. Error code: [v1], message: [v2], provider: [v3].

153 ADerrorFailedToChangePassword: Unable to reset the machine account password. Ensure the appropriate permissions exist to perform this

Active Directory task. Error code: [v1], message: [v2], provider: [v3].

154 ADerrorFailedToCopyDCName: Unable to copy the Domain Controller name. Error code: [v1], message: [v2], provider: [v3].

155 ADerrorDCNameIsTooLong: The Domain Controller name entered exceeds the maximum character length of [v4]. Error code: [v1], message: [v2], provider: [v3].

156 SiteMakUserPassword: The Site's makUser and makPassword fields must have values.

157 VamtError: See the log for additional error details.

158 InactiveDevice: Device specified is not active.

159 DiskIsInPrivateMode: This task cannot be performed because the vDisk is in private image mode.

160 AlreadyInChangeMode: Unable to complete this operation, vDisk is already in Maintenance, Merge, or Test mode.

161 CannotCreateMaintenanceDisk: Cannot create maintenance vDisk.

162 CannotEnterMaintenanceMode: To place a vDisk in Maintenance Mode requires using a Server. No Server is available at this time.

163 NotInMaintenanceMode: Unable to perform this action because the vDisk is not in Maintenance Mode.

164 NoVersionForMaintenanceMode: Unable to place this vDisk in Maintenance Mode because the highest version is not found.

165 NoVersionFound: Unable to perform this action because a version record was not found in the database.

166 Obsolete: The [v1] feature is obsolete.

167 DatabaseWarning: A database warning occurred.

168 DatabaseSQL: A database SQL error occurred.

169 DatabaseResource: A database resource error occurred.

170 InvalidUpdateTask: The specified UpdateTask does not exist.

171 InvalidVirtualHostingPool: The specified VirtualHostingPool does not exist.

172 RemoteCommand: An exception occurred executing a command on a remote Server.

173 IpcNotConfigured: An internal error occurred. The process communication interface must be configured before executing remote commands.

174 DiskAlreadSetForUpdate: The vDisk is already set for Update with Device [v1] in Site [v2].

175 InvalidDiskVersion: The vDisk Version specified is not valid.

176 HostResolution: Could not resolve the host name for [v1].

177 InProcess: The remote task is taking longer than expected. TaskId: [v1]

178 DateMustBeInFuture: The [v1] must be in the future.

179 InvalidRemoteReturn: The remote command did not return valid data.

180 InvalidParameterValueMinMax: The [v1] parameter value is invalid, the minimum is [v2] and the maximum is [v3].

181 InvalidParameterNotNumeric: The [v1] parameter value is invalid, it is not numeric.

182 InvalidParameterNotGuid: The [v1] parameter value is invalid, it is not a GUID.

183 PassThroughMessage: [v1]

184 DiskUpdateNotEnabled: The Automatic vDisk Update option must be enabled and the vDisk Update Server must be defined. Set these in the Site properties.

185 PvsStatusException:

Windows API error occurred, number 0xE000FFFF.

SQL error occurred, number 0xE001FFFF.

Manager error occurred. Error number 0xE002FFFF.

StreamProcess error occurred. Error number 0xE003FFFF.

Stream Database error occurred. Error number 0xE004FFFF.

Management error occurred. Error number 0xE005FFFF.

Shutdown in progress; request ignored. Error number 0xE0050001.

CreateDiffDisk: Malformed packet; missing one or more arguments. Error number 0xE0050002.

DeleteDiffDisk: Malformed file name; cannot parse directory and name. Error number 0xE0050003.

DeleteDiffDisk: Malformed packet; missing one or more arguments. Error number 0xE0050004.

IPC: Failed to read mtGetLocks parameters. Error number 0xE0050005.

IPC: Failed to read mtGetLockStatus parameters. Error number 0xE0050006.

IPC: Failed to read mtLock parameters. Error number 0xE0050007.

IPC: Failed to read mtUnlock parameters. Error number 0xE0050008.

MergeDisk event: Malformed packet; unknown message type. Error number 0xE0050009.

MergeDisk event: Unknown target request ID. Error number 0xE005000A.

MergeDisk event: Malformed packet; missing one or more arguments. Error number 0xE005000B.

MergeDisk: Malformed packet; missing one or more arguments. Error number 0xE005000C.

ValidateDisk: Malformed packet; missing one or more arguments. Error number 0xE005000D.

VHD Library error occurred. Error number 0xE006FFFF.

VHD Library: Not implemented. Error number 0xE0060001.

VHD Library: Handle pointer is invalid. Error number 0xE0060002.

VHD Library: Length of the path exceeds the limit of the file system. Error number 0xE0060003.

VHD Library: Name is empty. Error number 0xE0060004.

VHD Library: Length of the name exceeds the limit of the file system. Error number 0xE0060005.

VHD Library: Size of a parameter was too big. Error number 0xE0060006.

VHD Library: Size of a parameter was too small. Error number 0xE0060007.

VHD Library: The media is write protected. Error number 0xE0060008.

VHD Library: Type is invalid. Error number 0xE0060009.

VHD Library: Footer is incomplete. Error number 0xE006000A.

VHD Library: Failed to read or write the entire VHD Header. Error number 0xE006000B.

VHD Library: Failed to read or write the entire VHD Block Allocation Table. Error number 0xE006000C.

VHD Library: Failed to read or write all of the VHD properties. Error number 0xE006000D.

VHD Library: VHD footer is corrupt. Error number 0xE006000E.

VHD Library: VHD header is corrupt. Error number 0xE006000F.

VHD Library: Failed to read or write the VHD objects. Error number 0xE0060010.

VHD Library: Destination string is too small. Error number 0xE0060011.

VHD Library: Destination string pointer is NULL. Error number 0xE0060012.

VHD Library: Source string pointer is NULL. Error number 0xE0060013.

VHD Library: Offset is before the beginning of the VHD data area. Error number 0xE0060014.

VHD Library: Offset is after the end of the VHD data area. Error number 0xE0060015.

VHD Library: Failed to allocate memory because it was unavailable. Error number 0xE0060016.

VHD Library: Caller cancelled the last create request. Error number 0xE0060017.

VHD Library: Failed to read or write all of the data as requested. Error number 0xE0060018.

VHD Library: Failed to create a Universal Unique Identification for a VHD. Error number 0xE0060019.

VHD Library: Failed to find the VHD properties. Error number 0xE006001A.

VHD Library: Failed to read or write the entire sector bitmap within a block. Error number 0xE006001B.

VHD Library: Failed to read or write the entire block. Error number 0xE006001C.

VHD Library: Failed to open the file that represents the VHD. Error number 0xE006001D.

VHD Library: Requested number of bytes exceeds the remainder of bytes in a block. Error number 0xE006001E.

VHD Library: Accessed past end of the VHD file. Error number 0xE006001F.

VHD Library: Differencing VHD Unique ID (UUID) differs to parent VHD Unique ID. Error number 0xE0060020.

VHD Library: Differencing VHD timestamp differs to parent VHD last modified time. Error number 0xE0060021.

VHD Library: Failed to read or write the entire VHD Block Allocation Table Map. Error number 0xE0060022.

IPC error occurred. Error number 0xE007FFFF.

There was an unknown transmission error. Error number 0xE0070001.

No response received for successful send. Error number 0xA0070002.

Message processor timed out. Error number 0xE0070003.

Retry limit exhausted. Error number 0xE0070004.

Message recipient task is not active. Error number 0xE0070005.

Socket send/recv cannot be retried. Error number 0xE0070006.

Port shutdown due to connection opens exhausted. Error number 0xE0070007.

Port shutdown due to flood of junk packets. Error number 0xE0070008.

Port shutdown due to receive retries exhausted. Error number 0xE0070009.

Transport does not support fragmentation. Error number 0xE007000A.

One or more packet fragments are missing. Error number 0xE007000B.

Error sending message. Error number 0xE0070100.

Message acknowledgement timeout. Error number 0xA0070101.

Command timeout. Error number 0xE0070102.

Not implemented. Error number 0xE0070103.

Error verifying message port number, must be >= 0 and <= 65535. Error number 0xE0070104.

Command initialization failed. Error number 0xE0070105.

Start of IPC failed. Error number 0xE0070106.

Stop of IPC failed. Error number 0xE0070107.

Memory allocation failure. Error number 0xE0070108.

Internal error, failure to wait long enough for a communication response to be received. Error number 0xE0070109.

Disk Update error occurred. Error number 0xE008FFFF.

Inventory error occurred. Error number 0xE009FFFF.

Inventory Table: Failed to start thread. Error number 0xE0090001.

Inventory Table: Invalid Entry. Error number 0xE0090002.

Inventory Table: Failed to initialize inventory. Error number 0xE0090003.

Shutdown in progress; request ignored. Error number 0xE0090004.

Get Disk Inventory: Parameters bad. Error number 0xE0090033.

Populate database: Failed offline. Error number 0xE0090065.

Populate database: Server get by name failed. Error number 0xE0090066.

Populate database: Uninitialized. Error number 0xE0090067.

Populate database: Get host name failed. Error number 0xE0090068.

Populate database: Char conversion failed. Error number 0xE0090069.

Populate database: Initialization failed. Error number 0xE009006A.

Populate database: Database open failed. Error number 0xE009006B.

Populate database: Get all disk locators failed. Error number 0xE009006C.

Inventory Table: Not yet implemented. Error number 0xE009006D.

Notifier error occurred. Error number 0xE00AFFFF.

MAPI error occurred. Error number 0xE00BFFFF.

186 TaskCancelled: Task [v1] is cancelled and is not running.

187 TaskCompleted: Task [v1] has been completed and is not running.

188 TaskInProgress: Task [v1] is running and cannot be processed.

189 InvalidTask: The specified Task does not exist.

190 InventoryServerCannotContactDatabase: The Inventory Service cannot contact the database.

191 ServerOffline: The Server is offline.

192 ServerStateUnknown: The Server state is unknown.

193 HighestVersionIsPending: Could not complete this action because the   highest vDisk version is still pending. The scheduled date for the version has not occurred yet.

194 MergeInvalidWithCurrentVersions: Merge is not valid with the current versions that exist.

195 DiskInventoryError: vDisk versions are not up to date on all Servers that access this vDisk. Update all Servers with the latest versions of the vDisk files.

196 VDiskFileNotFoundWarning: [v1]: vDisk file was not found because it was deleted.

197 CannotAssignActiveServer: Stop the Server before attempting to assign the Server to a different site.

198 CannotAssignServerWithActiveDevice: Before attempting to assign the Server to a different site, shut down Devices connecting to the Server, then shut down the Server.

199 MappedDiskLocator: The vDisk is mapped and cannot be changed.

200 InvalidTemplateDevice: The Template Device must be a Production Device that does not have a Personal vDisk.

201 DeviceWithPersonalVDiskInvalid: Unable to process a Device that uses a personal vDisk.

202 CreatingDisk: Server is creating a vDisk so change cannot be done.

203 AssignedDiskLocatorToDeviceWithPersonalvDisk: Unable to delete a vDisk if the vDisk is currently assigned to a Device that uses a Personal vDisk. Unassign the Device, then delete the vDisk.

204 InvalidMacAddress: The MAC address for this VM is invalid. Configure the VM with a valid MAC address.

205 CannotGetMacFromHypervisor: The hypervisor did not return the MAC address for this VM: [v1]

206 Win32SystemException: A system error occurred.

207 RemoteManagementIpCannotBeResolved: Unable to resolve the management IP for Server [v1].

208 LocalManagementIpNotSet: The management IP for local server [v1] is not set in registry IPC\IPv4Address.

209 PerformVolumeMaintenanceTaskPermissions: Ensure the Service Account user has the appropriate 'Perform volume maintenance task' permissions.

210 CannotLoginToVirtualHostingPool: Unable to log on to the virtual hosting pool [v1]. Ensure that the hypervisior server is running properly.

211 VirtualHostingPoolNotSetForDevice: The virtualHostingPoolId for device [v1] with bdmBoot must be set.

212 ActiveBdmBootDeviceCannotProcess: The Boot Device Manager [v1] did not process successfully.

213 CannotMovePvdDeviceToAnotherSite: Personal vDisk Devices cannot be moved to another site.

214 XenDesktopSiteInvalid: XenDesktop Site for Devices is not valid, the XenDesktop Site is: [v1]

215 XenDesktopServiceListOutOfDate: XenDesktop Site [v1] is not reachable, check that the Citrix PVS Soap Server service user has XenDesktop permissions and network connectivity.

216 NoXenDesktopServiceForPersonalVDiskCapability: No XenDesktop service found for Personal vDisk capability.

217 InsufficientPermissionsToPreparePersonalVDisks: The user account for the Citrix PVS Soap Server has insufficient permissions to prepare Personal vDisks.

218 NotEnoughFreeDiskSpaceForManifest: There is not enough free disk space to create the manifest.

219 OperationCannotBeDoneOnlyPvdDevicesAssigned: Operation cannot be done, only Personal vDisk Devices are assigned.

220 DiskFormatCannotBeSetToVHD: The format cannot be set to VHD since no VHD vDisk file is found in the path, [v1], for Server, [v2].

221 DiskFormatCannotBeSetToVHDX: The format cannot be set to VHDX since no VHDX vDisk file is found in the path, [v1], for Server, [v2].

222 TemporaryVersionIsSet: This task cannot be performed because a temporary version is set.

223 DiskIsUsingPersistentCacheOnServer: A temporary version cannot be used for a vDisk that is using persistent cache on server.

224 UploadAlreadyInProgress: An upload is already in progress by Server [v1].

225 FieldMustBeNull: Field [v1] must be null.

226 DuplicateData: Record already exists in [v1] table for Farm.

227 CisUploadTokenGenerateError: Error generating upload token for My Citrix username [v1] ([v2]).

228 InvalidCredentials: The username or password is incorrect.

229 NoWriteAccessToFolders: No write access to folders [v1] or [v2].

230 ReportCreationError: Error creating problem report: [v1].

231 PvsProxyNotSupported: PVS Proxy not supported on this host

232 CannotCreateRegKey: Cannot create Registry key [v1]

4100 ADerrorUnexpectedError: An unexpected Active Directory related error occured. Ensure the appropriate permissions exist to perform this task. Error code: [v1], message: [v2], provider: [v3].

# Mcli-Add

```
Mcli-Add commandName[ -r field=value[, field2=value2]]
```

Mcli-Add AuthGroup

Add an Active Directory or Windows Group for authorization.

New-PvsAuthGroup

Create a new authorization AuthGroup for an Active Directory or Windows Group.

-r   Record to add. All record fields that do not have a Default are required.

All parameters that do not have a Default are required, unless only a few of a group are required.

authGroupName   Name of the Active Directory or Windows Group. Max Length=450

string Name or AuthGroupName: Name of the Active Directory or Windows Group. Max Length=450

description   User description. Default="" Max Length=250

string Description: User description. Default="" Max Length=250

Optional

SwitchParameter Confirm: The impact of this operation is "low". If -Confirm is specified, the operation will be confirmed. $ConfirmPreference can be set to "low" to have confirmation without the Confirm parameter.

resultId   If successful, the GUID of the record added.

PvsAuthGroup: If successful, the new PvsAuthGroup object is returned.

Mcli-Add CeipData

Add a new entry to CeipData table.

New-PvsCeipData

Create a new entry for CeipData table.

-r   Record to add. All record fields that do not
                            have a Default are required.

All parameters that do not have a Default are required, unless only a
                   few of a group are required.

enabled   1 if CEIP is enabled, otherwise 0. Min=0,
                            Max=1

uint Enabled: 1 if CEIP is enabled, otherwise 0. Min=0, Max=1

nextUpload   Date and time next CEIP upload is due if
                            enabled is 1. Default=null

DateTime NextUpload: Date and time next CEIP upload is due if enabled
                   is 1. Default=Empty

inProgress   1 if an upload is currently in progress,
                            otherwise 0. Default=0

uint InProgress: 1 if an upload is currently in progress, otherwise
                   0. Default=0

serverId   ID of server that is currently uploading,
                            null if inProgress is 0. Default=null
                            Length=36

Guid ServerId: ID of server that is currently uploading, null if
                   InProgress is 0. Default=00000000-0000-0000-0000-
                   000000000000

oneTimeUpload   1 to perform a one time upload. Default=0

uint OneTimeUpload: 1 to perform a one time upload. Default=0

Optional

SwitchParameter Confirm: The impact of this operation is "low". If -
                   Confirm is specified, the operation will be
                   confirmed. $ConfirmPreference can be set to "low" to
                   have confirmation without the Confirm parameter.

resultId   If successful, the GUID of the record added.

PvsCeipData: If successful, the new PvsCeipData object is returned.

Mcli-Add CisData

Add a new entry to CisData table.

New-PvsCisData

Create a new entry for CisData table.

-r   Record to add. All record fields that do not
                            have a Default are required.

All parameters that do not have a Default are required, unless only a few of a group are required.

userName   Username used to obtain the token Default=null Max Length=255

string UserName: Username used to obtain the token Max Length=255

path   Path where the last problem report bundle was saved Default=null Max Length=255

string Path: Path where the last problem report bundle was saved Default="" Max Length=255

password   Password of the user required to obtain the token. This is required only by Set and Add

string Password: Password of the user required to obtain the token. This is required only by Set and Add

Optional

SwitchParameter Confirm: The impact of this operation is "low". If -Confirm is specified, the operation will be confirmed. $ConfirmPreference can be set to "low" to have confirmation without the Confirm parameter.

resultId   If successful, the GUID of the record added.

PvsCisData: If successful, the new PvsCisData object is returned.

Mcli-Add Collection

Add a Collection to a Site.

New-PvsCollection

Create a new Collection for a Site.

-r   Record to add. All record fields that do not have a Default are required.

All parameters that do not have a Default are required, unless only a few of a group are required.

collectionName   Name of the Collection. It is unique within the Site. Max Length=50

string Name or CollectionName: Name of the Collection. It is unique within the Site. Max Length=50

siteId   GUID of the Site that this Collection is a member of. It is not used with siteName. Length=36

Guid SiteId: GUID of the Site that this Collection is a member of. It is not used with SiteName.

siteName   Name of the Site that this Collection is a member of. It is not used with siteId.

string SiteName: Name of the Site that this Collection is a member of. It is not used with SiteId.

description   User description. Default="" Max Length=250

string Description: User description. Default="" Max Length=250

lastAutoAddDeviceNumber   The Device Number of the last Auto Added
                           Device. Default=0

   *uint LastAutoAddDeviceNumber: The Device Number of the last Auto
              Added Device. Default=0*

enabled   1 when Devices in the Collection can be
                           booted, 0 otherwise. Default=1

  *SwitchParameter Disabled: If -Disabled is specified, the Devices in
              the Collection can not be booted. By default the
              Devices can be booted.*

autoAddPrefix   The string put before the Device Number for
                           Auto Add. Default=null ASCII computer name
                           characters no end digit Max Length=12

  *string AutoAddPrefix: The string put before the Device Number for
              Auto Add. Default="" ASCII computer name characters
              no end digit Max Length=12*

autoAddSuffix   The string put after the Device Number for
                           Auto Add. Default=null ASCII computer name
                           characters no begin digit Max Length=12

  *string AutoAddSuffix: The string put after the Device Number for Auto
              Add. Default="" ASCII computer name characters no
              begin digit Max Length=12*

autoAddZeroFill   1 when zeros be placed before the Device
                           Number up to the autoAddNumberLength for
                           Auto Add, 0 otherwise. Default=1

  *SwitchParameter NoAutoAddZeroFill: If -NoAutoAddZeroFill is
              specified, zeros will not be placed before the
              Device Number up to the AutoAddNumberLength for Auto
              Add.*

autoAddNumberLength   The maximum length of the Device Number for
                           Auto Add. This length plus the autoAddPrefix
                           length plus the autoAddSuffix length must be
                           less than 16. Required that
                           ((lenautoAddPrefix+lenautoAddSuffix)+autoAdd
                           NumberLength)<=15. Min=3, Max=9, Default=4

  *uint AutoAddNumberLength: The maximum length of the Device Number for
              Auto Add. This length plus the AutoAddPrefix length
              plus the AutoAddSuffix length must be less than 16.
              Required that
              ((lenautoAddPrefix+lenautoAddSuffix)+AutoAddNumberLe
              ngth)<=15. Min=3, Max=9, Default=4*

  Optional

  *SwitchParameter Confirm: The impact of this operation is "low". If -
              Confirm is specified, the operation will be
              confirmed. $ConfirmPreference can be set to "low" to
              have confirmation without the Confirm parameter.*

resultId   If successful, the GUID of the record added.

  PvsCollection: If successful, the new PvsCollection object is returned.

Mcli-Add Device

Add a new Device to a collection.

New-PvsDevice

Add a new Device to a Collection.

-r   Record to add. All record fields that do not
                            have a Default are required.

All parameters that do not have a Default are required, unless only a
few of a group are required.

deviceName   Computer name with no spaces. ASCII computer
                            name characters Max Length=15

string Name or DeviceName: Computer name with no spaces. ASCII
computer name characters Max Length=15

collectionId   GUID of the Collection this Device is to be
                            a member of. It is not used with
                            collectionName. Length=36

Guid CollectionId: GUID of the Collection this Device is to be a
member of. It is not used with CollectionName.

collectionName   Name of the Collection this Device is to be
                            a member of. siteName or siteId must also be
                            used.

string CollectionName: Name of the Collection this Device is to be a
member of. SiteName or SiteId must also be used.

siteId   GUID of the Site the collectionName is to be
                            a member of. This or siteName is used with
                            collectionName.

Guid SiteId: GUID of the Site the CollectionName is to be a member
of. This or SiteName is used with CollectionName.

siteName   Name of the Site the collectionName is to be
                            a member of. This or siteId is used with
                            collectionName.

string SiteName: Name of the Site the CollectionName is to be a
member of. This or SiteId is used with
CollectionName.

description   User description. Default="" Max Length=250

string Description: User description. Default="" Max Length=250

deviceMac   Ethernet address as a string in the form XX-
                            XX-XX-XX-XX-XX. Uniquely identifies the
                            Device. Length=17

PvsPhysicalAddress DeviceMac: Ethernet address can have the form XX-
XX-XX-XX-XX-XX. Uniquely identifies the Device.

bootFrom   Device to boot from. Choices are 1 for
                            vDisk, 2 for Hard Disk, and 3 for Floppy.
                            This cannot be Set for a Device with
                            Personal vDisk. Min=1, Max=3, Default=1

uint BootFrom: Device to boot from. Choices are 1 for vDisk, 2 for
                    Hard Disk, and 3 for Floppy. Min=1, Max=3, Default=1

className   Used by Automatic Update feature to match
                        new versions of Disks to a Device. This
                        cannot be Set for a Device with Personal
                        vDisk. Default="" Max Length=41

string ClassName: Used by Automatic Update feature to match new
                    versions of Disks to a Device. Default="" Max
                    Length=41

port   UDP port to use with Stream Service.
                        Min=1025, Max=65534, Default=6901

uint Port: UDP port to use with Stream Service. Min=1025, Max=65534,
                    Default=6901

enabled   1 when it can be booted, 0 otherwise. This
                        cannot be Set for a Device with Personal
                        vDisk. Default=1

SwitchParameter Disabled: If -Disabled is specified, the Device can
                    not be booted. By default the Device can be booted.

localDiskEnabled   If there is a local disk menu choice for the
                        Device, this is 1. This cannot be Set for a
                        Device with Personal vDisk. Default=0

SwitchParameter LocalDiskEnabled: If -LocalDiskEnabled is specified,
                    there will be a local disk menu choice for the
                    Device.

authentication   Device log in authentication. Choices are 0
                        for none, 1 for User Name/Password, and 2
                        for Extern. This cannot be Set for a Device
                        with Personal vDisk. Min=0, Max=2, Default=0

uint Authentication: Device log in authentication. Choices are 0 for
                    none, 1 for User Name/Password, and 2 for Extern.
                    Min=0, Max=2, Default=0

user   Name of user to authenticate before the boot
                        process continues. This cannot be Set for a
                        Device with Personal vDisk. Default="" ASCII
                        Max Length=20

string User: Name of user to authenticate before the boot process
                    continues. Default="" ASCII Max Length=20

password   Password of user to authenticate before the
                        boot process continues. This cannot be Set
                        for a Device with Personal vDisk. Default=""
                        ASCII Max Length=100

string Password: Password of user to authenticate before the boot
                    process continues. Default="" ASCII Max Length=100

copyTemplate   1 if the Template Device for the collection,
                        if it exists, should be used for the
                        property settings of the added Device.
                        Default=0

SwitchParameter CopyTemplate: If -CopyTemplate is specified, the Template Device for the collection, if it exists, will be used for the property settings of the new Device.

adTimestamp   The time the Active Directory machine account password was generated. Do not set this field, it is only set internally by PVS. Default=0

uint AdTimestamp: The time the Active Directory machine account password was generated. Do not set this field, it is only set internally by PVS. Default=0

adSignature   The signature of the Active Directory machine account password. Do not set this field, it is only set internally by PVS. Default=0

uint AdSignature: The signature of the Active Directory machine account password. Do not set this field, it is only set internally by PVS. Default=0

adPassword   The Active Directory machine account password. Do not set this field, it is only set internally by PVS. Default="" ASCII Max Length=256

string AdPassword: The Active Directory machine account password. Default="" ASCII Max Length=256

logLevel   Level to perform logging at. Values are: 0 (None), 1 (Fatal), 2 (Error), 3 (Warning), 4 (Info), 5 (Debug), and 6 (Trace). Min=0, Max=6, Default=0

uint LogLevel: Level to perform logging at. Values are: 0 (None), 1 (Fatal), 2 (Error), 3 (Warning), 4 (Info), 5 (Debug), and 6 (Trace). Min=0, Max=6, Default=0

domainName   Fully qualified name of the domain that the Device belongs to. Do not set this field, it is only set internally by PVS. Default=null Max Length=255

string DomainName: Fully qualified name of the domain that the Device belongs to. Do not set this field, it is only set internally by PVS. Default="" Max Length=255

domainObjectSID   The value of the objectSID AD attribute of the same name for the Device's computer account. Do not set this field, it is only set internally by PVS. Default=null Max Length=186

string DomainObjectSID: The value of the objectSID AD attribute of the same name for the Device's computer account. Do not set this field, it is only set internally by PVS. Default="" Max Length=186

domainControllerName   The name of the DC used to create the host's computer account. Do not set this field, it

is only set internally by PVS. Default=null
Max Length=4000

string DomainControllerName: The name of the DC used to create the host's computer account. Do not set this field, it is only set internally by PVS. Default="" Max Length=4000

domainTimeCreated   The time that the computer account was created. Do not set this field, it is only set internally by PVS. Format is: YYYY-MM-DD HH:MM:SS.SSS Default=null

DateTime DomainTimeCreated: The time that the computer account was created. Has the date and time including milliseconds. Do not set this field, it is only set internally by PVS. Default=Empty

type   1 when it performs test of Disks, 2 when it performs maintenance on Disks, 3 when it has a Personal vDisk, 4 when it has a Personal vDisk and performs tests, 0 otherwise. Min=0, Max=2, Default=0

uint Type: 1 when it performs test of Disks, 2 when it performs maintenance on Disks, 0 otherwise. Min=0, Max=2, Default=0

localWriteCacheDiskSize   The size in GB to format the Device cache file disk. If the value is 0, then the disk is not formatted. Min=0, Max=2048, Default=0

uint LocalWriteCacheDiskSize: The size in GB to format the Device cache file disk. If the value is 0, then the disk is not formatted. Min=0, Max=2048, Default=0

virtualHostingPoolId   GUID that uniquely identifies the Virtual Hosting Pool for a VM. This is needed when Adding a VM device. Default=null Length=36

Guid VirtualHostingPoolId: GUID that uniquely identifies the Virtual Hosting Pool for a VM. This is needed when Adding a VM device. Default=00000000-0000-0000-0000-000000000000

hypVmId   Hypervisor VM ID for HCL Default=null Max Length=250

string HypVmId: Hypervisor VM ID for HCL Default="" Max Length=250

bdmBoot   Use PXE boot when set to 0, BDM boot when set to 1. Default is PXE Default=0

SwitchParameter BdmBoot: If -BdmBoot is 0, use PXE, 1 use BDM. PXE boot is used by default.

bdmType   Use PXE boot when set to 0, BDM (Bios) boot when set to 1 and BDM (Uefi) boot when set to 2. Default=0

uint BdmType: Use PXE boot when set to 0, BDM (Bios) boot when set to 1 and BDM (Uefi) boot when set to 2. Default=0

```
    bdmFormat                      1 use VHD for BDMboot, 2 use ISO, 3 use USB.
                                   Default=0
```

uint BdmFormat: 1 use VHD for BDMboot, 2 use ISO, 3 use USB.
                Default=0

```
    bdmUpdated                     Timestamp of the last BDM boot disk update.
                                   Default=null
```

DateTime BdmUpdated: Timestamp of the last BDM boot disk update.
                Default=Empty

```
    bdmCreated                     Timstamp when BDM device was
                                   created  Default=null
```

DateTime BdmCreated: Timstamp when BDM device was
                created Default=Empty

```
    xsPvsProxyUuid                 UUID of XenServer PVS_proxy Default=null
                                   Length=36
```

Guid XsPvsProxyUuid: UUID of XenServer PVS_proxy Default=00000000-
                0000-0000-0000-000000000000

```
    enableXsProxy                  Enable XenServerProxy when set to 1
                                   Default=null
```

string EnableXsProxy: Enable XenServerProxy when set to 1 Default=""

Optional

SwitchParameter Confirm: The impact of this operation is "low". If -
                Confirm is specified, the operation will be
                confirmed. $ConfirmPreference can be set to "low" to
                have confirmation without the Confirm parameter.

```
  resultId                       If successful, the GUID of the record added.
```

PvsDevice: If successful, the new PvsDevice object is returned.

Mcli-Add DeviceWithPersonalvDisk

```
    Add a new Device with Personal vDisk to a collection.
```

New-PvsDeviceWithPersonalvDisk

Add a new Device with Personal vDisk to a collection.

```
    -r                             Record to add. All record fields that do not
                                   have a Default are required.
```

All parameters that do not have a Default are required, unless only a
                few of a group are required.

```
    deviceName                     Computer name with no spaces. ASCII computer
                                   name characters Max Length=15
```

string Name or DeviceName: Computer name with no spaces. ASCII
                computer name characters Max Length=15

```
    collectionId                   GUID of the Collection this Device with
                                   Personal vDisk is to be a member of. It is
                                   not used with collectionName. Length=36
```

collectionName   Name of the Collection this Device with Personal vDisk is to be a member of. siteName or siteId must also be used.

diskLocatorId   GUID of the Disk Locator to update with this Device.

siteId   GUID of the Site the collectionName is to be a member of. This or siteName is used with collectionName.

siteName   Name of the Site the collectionName is to be a member of. This or siteId is used with collectionName.

description   User description. Default="" Max Length=250

deviceMac   Ethernet address as a string in the form XX-XX-XX-XX-XX-XX. Uniquely identifies the Device with Personal vDisk. Length=17

port   UDP port to use with Stream Service. Min=1025, Max=65534, Default=6901

adTimestamp   The time the Active Directory machine account password was generated. Do not set this field, it is only set internally by PVS. Default=0

adSignature   The signature of the Active Directory machine account password. Do not set this field, it is only set internally by PVS. Default=0

uint AdSignature: The signature of the Active Directory machine
                  account password. Do not set this field, it is only
                  set internally by PVS. Default=0

logLevel   Level to perform logging at. Values are: 0
                                (None), 1 (Fatal), 2 (Error), 3 (Warning), 4
                                (Info), 5 (Debug), and 6 (Trace). Min=0,
                                Max=6, Default=0

uint LogLevel: Level to perform logging at. Values are: 0 (None), 1
                  (Fatal), 2 (Error), 3 (Warning), 4 (Info), 5
                  (Debug), and 6 (Trace). Min=0, Max=6, Default=0

domainName   Fully qualified name of the domain that the
                                Device with Personal vDisk belongs to. Do
                                not set this field, it is only set
                                internally by PVS. Default=null Max
                                Length=255

string DomainName: Fully qualified name of the domain that the Device
                  with Personal vDisk belongs to. Do not set this
                  field, it is only set internally by PVS. Default=""
                  Max Length=255

domainObjectSID   The value of the objectSID AD attribute of
                                the same name for the Device with Personal
                                vDisk's computer account. Do not set this
                                field, it is only set internally by PVS.
                                Default=null Max Length=186

string DomainObjectSID: The value of the objectSID AD attribute of
                  the same name for the Device with Personal vDisk's
                  computer account. Do not set this field, it is only
                  set internally by PVS. Default="" Max Length=186

domainControllerName   The name of the DC used to create the host's
                                computer account. Do not set this field, it
                                is only set internally by PVS. Default=null
                                Max Length=4000

string DomainControllerName: The name of the DC used to create the
                  host's computer account. Do not set this field, it
                  is only set internally by PVS. Default="" Max
                  Length=4000

domainTimeCreated   The time that the computer account was
                                created. Do not set this field, it is only
                                set internally by PVS. Format is: YYYY-MM-DD
                                HH:MM:SS.SSS Default=null

DateTime DomainTimeCreated: The time that the computer account was
                  created. Has the date and time including
                  milliseconds. Do not set this field, it is only set
                  internally by PVS. Default=Empty

type   3 when it has a Personal vDisk, 4 when it
                                has a Personal vDisk and performs tests.
                                Min=3, Max=4, Default=3

uint Type: 3 when it has a Personal vDisk, 4 when it has a Personal
                  vDisk and performs tests. Min=3, Max=4, Default=3

```
pvdDriveLetter              Personal vDisk Drive letter. Range is F to
                            Z. Default=null Max Length=1
```

string PvdDriveLetter: Personal vDisk Drive letter. Range is F to Z.
Default="" Max Length=1

```
localWriteCacheDiskSize   The size in GB to format the Device cache
                          file disk. If the value is 0, then the disk
                          is not formatted. Min=0, Max=2048, Default=0
```

uint LocalWriteCacheDiskSize: The size in GB to format the Device
cache file disk. If the value is 0, then the disk is
not formatted. Min=0, Max=2048, Default=0

```
bdmBoot                   Use BDM instead of PXE boot when set to 1.
                          Default=0
```

SwitchParameter BdmBoot: If -BdmBoot is specified, user BDM instead
of PXE boot. PXE boot is used by default.

```
xdSiteId                  GUID of the XenDesktop Site. Default=null
                          Length=36
```

Guid XdSiteId: GUID of the XenDesktop Site. Default=00000000-0000-
0000-0000-000000000000

```
xdCatalogId               Integer identifier of the XenDesktop
                          Catalog. Default=null
```

uint XdCatalogId: Integer identifier of the XenDesktop Catalog.
Default=""

```
virtualHostingPoolId      GUID that uniquely identifies the Virtual
                          Hosting Pool for a VM. This is needed when
                          Adding a VM device. Default=null Length=36
```

Guid VirtualHostingPoolId: GUID that uniquely identifies the Virtual
Hosting Pool for a VM. This is needed when Adding a
VM device. Default=00000000-0000-0000-0000-
000000000000

```
enableXsProxy             Enable XenServerProxy when set to 1
                          Default=null
```

string EnableXsProxy: Enable XenServerProxy when set to 1 Default=""

Optional

SwitchParameter Confirm: The impact of this operation is "low". If -
Confirm is specified, the operation will be
confirmed. $ConfirmPreference can be set to "low" to
have confirmation without the Confirm parameter.

```
resultId                     If successful, the GUID of the record added.
```

PvsDeviceWithPersonalvDisk: If successful, the new
PvsDeviceWithPersonalvDisk object is returned.

Mcli-Add DiskLocator

Add a Disk Locator to a Site. The Disk file must already exist.

New-PvsDiskLocator

Create a new Disk Locator for a Site. The Disk file must already exist.

-r   Record to add. All record fields that do not
                              have a Default are required.

   All parameters that do not have a Default are required, unless only a
                    few of a group are required.

   diskLocatorName   Name of the Disk Locator File. It is unique
                              within the Store. ASCII Max Length=52

      string Name or DiskLocatorName: Name of the Disk Locator File. It is
                    unique within the Store. ASCII Max Length=52

   siteId   GUID of the Site this DiskLocator is to be a
                              member of. It is not used with siteName.
                              Length=36

      Guid SiteId: GUID of the Site this DiskLocator is to be a member of.
                    It is not used with SiteName.

   siteName   Name of the Site this DiskLocator is to be a
                              member of. It is not used with siteId.

      string SiteName: Name of the Site this DiskLocator is to be a member
                    of. It is not used with SiteId.

   storeId   GUID of the Store that this Disk Locator is
                              a member of. siteName or siteId must also be
                              used. It is not used with storeName.
                              Length=36

      Guid StoreId: GUID of the Store that this Disk Locator is a member
                    of. SiteName or SiteId must also be used. It is not
                    used with StoreName.

   storeName   Name of the Store that this Disk Locator is
                              a member of. siteName or siteId must also be
                              used. It is not used with storeId.

      string StoreName: Name of the Store that this Disk Locator is a
                    member of. SiteName or SiteId must also be used. It
                    is not used with StoreId.

   description   User description. Default="" Max Length=250

      string Description: User description. Default="" Max Length=250

   menuText   Text that is displayed in the Boot Menu. If
                              this field has no value, the name value is
                              used. Default="" ASCII Max Length=64

      string MenuText: Text that is displayed in the Boot Menu. If this
                    field has no value, the name value is used.
                    Default="" ASCII Max Length=64

   serverId   GUID of the single Server that this Disk
                              Locator is assigned to. It is not used with
                              serverName. Default=null Length=36

      Guid ServerId: GUID of the single Server that this Disk Locator is
                    assigned to. It is not used with ServerName.
                    Default=00000000-0000-0000-0000-000000000000

   serverName   Name of the single Server that this Disk
                              Locator is assigned to. It is not used with
                              serverId. Default=null

    enabled   1 when this disk can be booted, 0 otherwise.
                            Default=1

    rebalanceEnabled   1 when this Server can automatically
                            rebalance Devices, 0 otherwise. Default=0

    rebalanceTriggerPercent   Percent over fair load that triggers a
                              dynamic Device rebalance. Min=5, Max=5000,
                              Default=25

    subnetAffinity   Qualifier for subnet affinity when assigning
                            a Server. 0=None, 1=Best Effort, 2=Fixed.
                            Min=0, Max=2, Default=0

    newDiskWriteCacheType   The writeCacheType that if a new Disk will
                              be created, it will be set with. It is only
                              used when a new Disk is being created. Value
                              are: 0 (Private), (other values are standard
                              image) 1 (Cache on Server), 3 (Cache in
                              Device RAM), 4 (Cache on Device Hard Disk),
                              7 (Cache on Server, Persistent), or 9 (Cache
                              in Device RAM with Overflow on Hard Disk)
                              Default=0

    format   Format of the image when adding a
                            DiskLocator for a vDisk that has never been
                            added to the Farm. VHDX is the only format
                            that requires this field. Values are: 0
                            (VHD) and 1 (VHDX). Default=0

resultId   If successful, the GUID of the record added.

Mcli-Add DiskUpdateDevice

Add a new Device related to a Disk that can be updated.

New-PvsDiskUpdateDevice

Add a new Device related to a Disk that can be updated.

-r   Record to add. All record fields that do not have a Default are required.

All parameters that do not have a Default are required, unless only a few of a group are required.

deviceName   Computer name with no spaces. ASCII computer name characters Max Length=15

string Name or DeviceName: Computer name with no spaces. ASCII computer name characters Max Length=15

virtualHostingPoolId   GUID of the Virtual Hosting Pool. It is not used with virtualHostingPoolName. Default=null Length=36

Guid VirtualHostingPoolId: GUID of the Virtual Hosting Pool. It is not used with VirtualHostingPoolName.

virtualHostingPoolName   Name of the Virtual Hosting Pool.

string VirtualHostingPoolName: Name of the Virtual Hosting Pool.

diskLocatorId   GUID of the Disk Locator to update with this Device.

Guid DiskLocatorId: GUID of the Disk Locator to update with this Device.

description   User description. Default="" Max Length=250

string Description: User description. Default="" Max Length=250

deviceMac   Ethernet address as a string in the form XX-XX-XX-XX-XX-XX. Uniquely identifies the Device. Length=17

PvsPhysicalAddress DeviceMac: Ethernet address can have the form XX-XX-XX-XX-XX-XX. Uniquely identifies the Device.

port   UDP port to use with Stream Service. Min=1025, Max=65534, Default=6901

uint Port: UDP port to use with Stream Service. Min=1025, Max=65534, Default=6901

adTimestamp   The time the Active Directory machine account password was generated. Do not set

this field, it is only set internally by
                            PVS. Default=0

  uint AdTimestamp: The time the Active Directory machine account
                password was generated. Do not set this field, it is
                only set internally by PVS. Default=0

adSignature   The signature of the Active Directory
                            machine account password. Do not set this
                            field, it is only set internally by PVS.
                            Default=0

  uint AdSignature: The signature of the Active Directory machine
                account password. Do not set this field, it is only
                set internally by PVS. Default=0

logLevel   Level to perform logging at. Values are: 0
                            (None), 1 (Fatal), 2 (Error), 3 (Warning), 4
                            (Info), 5 (Debug), and 6 (Trace). Min=0,
                            Max=6, Default=0

  uint LogLevel: Level to perform logging at. Values are: 0 (None), 1
                (Fatal), 2 (Error), 3 (Warning), 4 (Info), 5
                (Debug), and 6 (Trace). Min=0, Max=6, Default=0

domainName   Fully qualified name of the domain that the
                            Device belongs to. Do not set this field, it
                            is only set internally by PVS. Default=null
                            Max Length=255

  string DomainName: Fully qualified name of the domain that the Device
                belongs to. Do not set this field, it is only set
                internally by PVS. Default="" Max Length=255

domainObjectSID   The value of the objectSID AD attribute of
                            the same name for the Device's computer
                            account. Do not set this field, it is only
                            set internally by PVS. Default=null Max
                            Length=186

  string DomainObjectSID: The value of the objectSID AD attribute of
                the same name for the Device's computer account. Do
                not set this field, it is only set internally by
                PVS. Default="" Max Length=186

domainControllerName   The name of the DC used to create the host's
                            computer account. Do not set this field, it
                            is only set internally by PVS. Default=null
                            Max Length=4000

  string DomainControllerName: The name of the DC used to create the
                host's computer account. Do not set this field, it
                is only set internally by PVS. Default="" Max
                Length=4000

domainTimeCreated   The time that the computer account was
                            created. Do not set this field, it is only
                            set internally by PVS. Format is: YYYY-MM-DD
                            HH:MM:SS.SSS Default=null

  DateTime DomainTimeCreated: The time that the computer account was
                created. Has the date and time including

adPassword   The Active Directory machine account password. Do not set this field, it is only set internally by PVS. Default="" Max Length=256

string AdPassword: The Active Directory machine account password. Default="" Max Length=256

Optional

SwitchParameter Confirm: The impact of this operation is "low". If -Confirm is specified, the operation will be confirmed. $ConfirmPreference can be set to "low" to have confirmation without the Confirm parameter.

resultId   If successful, the GUID of the record added.

PvsDiskUpdateDevice: If successful, the new PvsDiskUpdateDevice object is returned.

Mcli-Add FarmView

Add a View to a Farm.

New-PvsFarmView

Create a new View for the Farm.

-r   Record to add. All record fields that do not have a Default are required.

All parameters that do not have a Default are required, unless only a few of a group are required.

farmViewName   name of the Farm View. Max Length=50

string Name or FarmViewName: name of the Farm View. Max Length=50

description   User description. Default="" Max Length=250

string Description: User description. Default="" Max Length=250

Optional

SwitchParameter Confirm: The impact of this operation is "low". If -Confirm is specified, the operation will be confirmed. $ConfirmPreference can be set to "low" to have confirmation without the Confirm parameter.

resultId   If successful, the GUID of the record added.

PvsFarmView: If successful, the new PvsFarmView object is returned.

Mcli-Add Server

Add a Server to a Site.

New-PvsServer

Add a new Server to a Site.

-r   Record to add. All record fields that do not have a Default are required.

serverName   Computer name with no spaces. ASCII computer
                                name characters Max Length=21

  string Name or ServerName: Computer name with no spaces. ASCII
                 computer name characters Max Length=21

siteId   GUID of the Site this Server is to be a
                                member of. It is not used with siteName.
                                Length=36

  Guid SiteId: GUID of the Site this Server is to be a member of. It is
                 not used with SiteName.

siteName   Name of the Site this Server is to be a
                                member of. It is not used with siteId.

  string SiteName: Name of the Site this Server is to be a member of.
                 It is not used with SiteId.

description   User description. Default="" Max Length=250

  string Description: User description. Default="" Max Length=250

adMaxPasswordAge   Number of days before a password expires.
                                Min=1, Max=30, Default=7

  uint AdMaxPasswordAge: Number of days before a password expires.
                 Min=1, Max=30, Default=7

licenseTimeout   Amount of seconds before a license times
                                out. Min=15, Max=300, Default=30

  uint LicenseTimeout: Amount of seconds before a license times out.
                 Min=15, Max=300, Default=30

vDiskCreatePacing   VDisk create time pacing in miliseconds.
                                Min=0, Max=5, Default=0

  uint VDiskCreatePacing: VDisk create time pacing in miliseconds.
                 Min=0, Max=5, Default=0

firstPort   Number of the first UDP port for use by the
                                Stream Service, First and Last must allow at
                                least 5 ports. Min=1025, Max=65534,
                                Default=6910

  uint FirstPort: Number of the first UDP port for use by the Stream
                 Service, First and Last must allow at least 5 ports.
                 Min=1025, Max=65534, Default=6910

lastPort   Number of the last UDP port for use by the
                                Stream Service, First and Last must allow at
                                least 5 ports. Min=1025, Max=65534,
                                Default=6930

  uint LastPort: Number of the last UDP port for use by the Stream
                 Service, First and Last must allow at least 5 ports.
                 Min=1025, Max=65534, Default=6930

threadsPerPort   Number of worker threads per IO port.
                                Required that (threadPerPort * numberPorts *
                                numberIPs) <= 1000. Min=1, Max=60, Default=8

uint ThreadsPerPort: Number of worker threads per IO port. Required that (threadPerPort * numberPorts * numberIPs) <= 1000. Min=1, Max=60, Default=8

buffersPerThread   Number of buffers per worker thread. Min=1, Max=128, Default=24

uint BuffersPerThread: Number of buffers per worker thread. Min=1, Max=128, Default=24

serverCacheTimeout   Number of seconds to wait before considering another Server is down. Min=5, Max=60, Default=8

uint ServerCacheTimeout: Number of seconds to wait before considering another Server is down. Min=5, Max=60, Default=8

ioBurstSize   Number of bytes read/writes can send in a burst of packets. Required that ioBurstSize/(maxTransmissionUnits-76)<=32. Min=4096, Max=61440, Default=32768

uint IoBurstSize: Number of bytes read/writes can send in a burst of packets. Required that IoBurstSize/(MaxTransmissionUnits-76)<=32. Min=4096, Max=61440, Default=32768

maxTransmissionUnits   Ethernet maximum transmission unit size for the protocol for use for Server and Device. Required that ioBurstSize/(maxTransmissionUnits-76)<=32. Min=502, Max=16426, Default=1506

uint MaxTransmissionUnits: Ethernet maximum transmission unit size for the protocol for use for Server and Device. Required that IoBurstSize/(MaxTransmissionUnits-76)<=32. Min=502, Max=16426, Default=1506

maxBootDevicesAllowed   Maximum number of Devices allowed to boot simultaneously. Min=1, Max=1000, Default=500

uint MaxBootDevicesAllowed: Maximum number of Devices allowed to boot simultaneously. Min=1, Max=1000, Default=500

maxBootSeconds   Maximum number of seconds for a Device to boot. Min=10, Max=900, Default=60

uint MaxBootSeconds: Maximum number of seconds for a Device to boot. Min=10, Max=900, Default=60

bootPauseSeconds   Number of seconds that a Device will pause during login if its server busy. Min=1, Max=60, Default=10

uint BootPauseSeconds: Number of seconds that a Device will pause during login if its server busy. Min=1, Max=60, Default=10

adMaxPasswordAgeEnabled   Age the password, when set to 1. Default=0

SwitchParameter AdMaxPasswordAgeEnabled: If -AdMaxPasswordAgeEnabled is specified, Age the password.

eventLoggingEnabled   Enable event logging, when set to 1. Default=0

nonBlockingIoEnabled   Use non-Blocking IO, when set to 1.
                            Default=1

ip   One or more streaming ip addresses. If more
                            than one ip is included, the ip addresses
                            are comma delimited.

initialQueryConnectionPoolSize Initial size of database connection
                            pool for non-transactional queries. Min=1,
                            Max=1000, Default=50

initialTransactionConnectionPoolSize Initial size of database
                            connection pool for transactional queries.
                            Min=1, Max=1000, Default=50

maxQueryConnectionPoolSize Maximum size of database connection pool
                            for non-transactional queries. Min=1,
                            Max=32767, Default=1000

maxTransactionConnectionPoolSize Maximum size of database connection
                            pool for transactional queries. Min=1,
                            Max=32767, Default=1000

refreshInterval   Interval, in number of seconds, the server
                            should wait before refreshing settings. If
                            set to 0, unused database connections are
                            never released. Min=0, Max=32767,
                            Default=300

unusedDbConnectionTimeout   Interval, in number of seconds, a
                            connection should go unused before it is to
                            be released. Min=0, Max=32767, Default=300

uint UnusedDbConnectionTimeout: Interval, in number of seconds, a
                    connection should go unused before it is to be
                    released. Min=0, Max=32767, Default=300

busyDbConnectionRetryCount   Number of times a failed database
                    connection will be retried. Min=0,
                    Max=32767, Default=2

uint BusyDbConnectionRetryCount: Number of times a failed database
                    connection will be retried. Min=0, Max=32767,
                    Default=2

busyDbConnectionRetryInterval   Interval, in number of milliseconds,
                    the server should wait before retrying to
                    connect to a database. Min=0, Max=10000,
                    Default=25

uint BusyDbConnectionRetryInterval: Interval, in number of
                    milliseconds, the server should wait before retrying
                    to connect to a database. Min=0, Max=10000,
                    Default=25

localConcurrentIoLimit   Maximum concurrent IO transactions it
                    performs for vDisks that are local. A value
                    of 0 disables the feature. Min=0, Max=128,
                    Default=4

uint LocalConcurrentIoLimit: Maximum concurrent IO transactions it
                    performs for vDisks that are local. A value of 0
                    disables the feature. Min=0, Max=128, Default=4

remoteConcurrentIoLimit   Maximum concurrent IO transactions it
                    performs for vDisks that are remote. A value
                    of 0 disables the feature. Min=0, Max=128,
                    Default=4

uint RemoteConcurrentIoLimit: Maximum concurrent IO transactions it
                    performs for vDisks that are remote. A value of 0
                    disables the feature. Min=0, Max=128, Default=4

logLevel   Level to perform logging at. Values are: 0
                    (None), 1 (Fatal), 2 (Error), 3 (Warning), 4
                    (Info), 5 (Debug), and 6 (Trace). Min=0,
                    Max=6, Default=4

uint LogLevel: Level to perform logging at. Values are: 0 (None), 1
                    (Fatal), 2 (Error), 3 (Warning), 4 (Info), 5
                    (Debug), and 6 (Trace). Min=0, Max=6, Default=4

logFileSizeMax   Maximum size log files can reach in
                    Megabytes. Min=1, Max=50, Default=5

uint LogFileSizeMax: Maximum size log files can reach in Megabytes.
                    Min=1, Max=50, Default=5

logFileBackupCopiesMax   Maximum number of log file backups. Min=1,
                    Max=50, Default=4

uint LogFileBackupCopiesMax: Maximum number of log file backups.
                    Min=1, Max=50, Default=4

powerRating   A strictly relative rating of this Server's
                    capabilities when compared to other Servers
                    in the Store(s) it belongs too; can be used

to help tune load balancing. Min=0.1,
Max=1000, Default=1

float PowerRating: A strictly relative rating of this Server's
capabilities when compared to other Servers in the
Store(s) it belongs too; can be used to help tune
load balancing. Min=0.1, Max=1000, Default=1

lastCeipUploadAttempt   Time that this server last attempted a CEIP
upload. Default=null

DateTime LastCeipUploadAttempt: Time that this server last attempted
a CEIP upload. Default=Empty

lastBugReportAttempt   Time that this server last attempted to
upload or generate a bug report bundle.
Default=null

DateTime LastBugReportAttempt: Time that this server last attempted
to upload or generate a bug report bundle.
Default=Empty

lastBugReportStatus   Status of the last bug report on this
server. Default=null Max Length=250

string LastBugReportStatus: Status of the last bug report on this
server. Default="" Max Length=250

lastBugReportResult   Status of the last bug report on this
server. Default=null Max Length=4000

string LastBugReportResult: Status of the last bug report on this
server. Default="" Max Length=4000

lastBugReportSummary   Summary of the last bug report on this
server. Default=null Max Length=250

string LastBugReportSummary: Summary of the last bug report on this
server. Default="" Max Length=250

Optional

SwitchParameter Confirm: The impact of this operation is "low". If -
Confirm is specified, the operation will be
confirmed. $ConfirmPreference can be set to "low" to
have confirmation without the Confirm parameter.

resultId   If successful, the GUID of the record added.

PvsServer: If successful, the new PvsServer object is returned.

Mcli-Add Site

Add a Site to the system.

New-PvsSite

Create a new Site for the Farm.

-r   Record to add. All record fields that do not
have a Default are required.

All parameters that do not have a Default are required, unless only a
few of a group are required.

siteName   Name of the Site. Max Length=50

   string Name or SiteName: Name of the Site. Max Length=50

description   User description. Default="" Max Length=250

   string Description: User description. Default="" Max Length=250

inventoryFilePollingInterval The number of seconds between polls for
                           Disk changes in the Stores. Min=1, Max=600,
                           Default=60

   uint InventoryFilePollingInterval: The number of seconds between
                polls for Disk changes in the Stores. Min=1,
                Max=600, Default=60

enableDiskUpdate   1 when Disk Updated is enabled for the Site,
                       0 otherwise. Default=0

  SwitchParameter EnableDiskUpdate: If -EnableDiskUpdate is specified,
                the Disk Update will be enabled for the Site. By
                default Disk Update is disabled.

diskUpdateServerId   GUID of the Disk Update Server for the Site.
                       Not used with diskUpdateServerName.
                       Default=null Length=36

  Guid DiskUpdateServerId: GUID of the Disk Update Server for the Site.
                Not used with DiskUpdateServerName.
                Default=00000000-0000-0000-0000-000000000000

diskUpdateServerName   Name of the Disk Update Server for the Site.
                       Not used with diskUpdateServerId.
                       Default=null

   string DiskUpdateServerName: Name of the Disk Update Server for the
                Site. Not used with DiskUpdateServerId. Default=""

makUser   User name used for MAK activation.
                       Default=null Max Length=64

   string MakUser: User name used for MAK activation. Default="" Max
                Length=64

makPassword   User password used for MAK activation.
                       Default=null Max Length=64

   string MakPassword: User password used for MAK activation. Default=""
                Max Length=64

enableXsProxy   Enable XenServerProxy when set to 1
                     Default=null

   string EnableXsProxy: Enable XenServerProxy when set to 1 Default=""

virtualHostingPoolId   GUID of the VirtualHostingPool object.

  Guid VirtualHostingPoolId: GUID of the VirtualHostingPool object.

virtualHostingPoolName   Name of the VirtualHostingPool object.

   string VirtualHostingPoolName: Name of the VirtualHostingPool object.

xsPvsSiteUuid   GUID of the XenServer PVS Site.

   string XsPvsSiteUuid: GUID of the XenServer PVS Site.

Optional

         resultId   If successful, the GUID of the record added.

         PvsSite: If successful, the new PvsSite object is returned.

Mcli-Add SiteView

Add a View to a Site.

New-PvsSiteView

Create a new View for a Site.

         -r   Record to add. All record fields that do not
                                    have a Default are required.

       All parameters that do not have a Default are required, unless only a
                        few of a group are required.

       siteViewName   Name of the Site View. Max Length=50

         string Name or SiteViewName: Name of the Site View. Max Length=50

       siteId   GUID of the Site this View is to be a member
                                    of. It is not used with siteName. Length=36

         Guid SiteId: GUID of the Site this View is to be a member of. It is
                        not used with SiteName.

       siteName   Name of the Site this View is to be a member
                                    of. It is not used with siteId.

         string SiteName: Name of the Site this View is to be a member of. It
                        is not used with SiteId.

       description   User description. Default="" Max Length=250

         string Description: User description. Default="" Max Length=250

       Optional

         SwitchParameter Confirm: The impact of this operation is "low". If -
                        Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "low" to
                        have confirmation without the Confirm parameter.

         resultId   If successful, the GUID of the record added.

         PvsSiteView: If successful, the new PvsSiteView object is returned.

Mcli-Add Store

Add a Store to the system.

New-PvsStore

Create a new Store for the Farm.

         -r   Record to add. All record fields that do not
                                    have a Default are required.

       All parameters that do not have a Default are required, unless only a
                        few of a group are required.

storeName   Name of the Store. Max Length=50

   string Name or StoreName: Name of the Store. Max Length=50

siteId   GUID of the Site where Administrators of
                               that Site can change this Store. Not used
                               for Farm Stores. siteName can be used
                               instead. Default=null Length=36

   Guid SiteId: GUID of the Site where Administrators of that Site can
                   change this Store. Not used for Farm Stores.
                   SiteName can be used instead. Default=00000000-0000-
                   0000-0000-000000000000

siteName   Name of the Site where Administrators of
                               that Site can change this Store. Not used
                               for Farm Stores. siteId can be used instead.
                               Default=null

   string SiteName: Name of the Site where Administrators of that Site
                   can change this Store. Not used for Farm Stores.
                   SiteId can be used instead. Default=""

description   User description. Default="" Max Length=250

   string Description: User description. Default="" Max Length=250

path   Default directory path that the Servers use
                               to access this Store. Max Length=255

   string Path: Default directory path that the Servers use to access
                   this Store. Max Length=255

cachePath   Default Cache path(s) that the Servers use
                               with this Store. It is an ordered comma-
                               delimited list. If none are specified the
                               caches will be placed in the WriteCache
                               subdirectory of the Store path. Default=None

   string[] CachePath: Default Cache path(s) that the Servers use with
                   this Store. If none are specified the caches will be
                   placed in the WriteCache subdirectory of the Store
                   path. Default=None

   Optional

   SwitchParameter Confirm: The impact of this operation is "low". If -
                   Confirm is specified, the operation will be
                   confirmed. $ConfirmPreference can be set to "low" to
                   have confirmation without the Confirm parameter.

resultId   If successful, the GUID of the record added.

   PvsStore: If successful, the new PvsStore object is returned.

Mcli-Add UpdateTask

Add a new Update Task for the Site.

New-PvsUpdateTask

Create a new Update Task for a Store.

-r   Record to add. All record fields that do not
                               have a Default are required.

All parameters that do not have a Default are required, unless only a few of a group are required.

updateTaskName	Name of the Update Task. It is unique within the Site. Max Length=50

string Name or UpdateTaskName: Name of the Update Task. It is unique within the Site. Max Length=50

siteId	GUID of the Site that this Update Task is a member of. It is not used with siteName. Length=36

Guid SiteId: GUID of the Site that this Update Task is a member of. It is not used with SiteName.

siteName	Name of the Site that this Update Task is a member of. It is not used with siteId.

string SiteName: Name of the Site that this Update Task is a member of. It is not used with SiteId.

description	User description. Default="" Max Length=250

string Description: User description. Default="" Max Length=250

enabled	1 when it will be processed, 0 otherwise. Default=1

SwitchParameter Disabled: If -Disabled is specified, the updates will not be processed. By default the updates will be processed.

hour	The hour of the day to perform the task. Min=0, Max=23, Default=0

uint Hour: The hour of the day to perform the task. Min=0, Max=23, Default=0

minute	The minute of the hour to perform the task. Min=0, Max=59, Default=0

uint Minute: The minute of the hour to perform the task. Min=0, Max=59, Default=0

recurrence	The update will reoccur on this schedule. 0 = None, 1 = Daily, 2 = Every Weekday, 3 = Weekly, 4 = Monthly Date, 5 = Monthly Type. Min=0, Max=5, Default=0

uint Recurrence: The update will reoccur on this schedule. 0 = None, 1 = Daily, 2 = Every Weekday, 3 = Weekly, 4 = Monthly Date, 5 = Monthly Type. Min=0, Max=5, Default=0

dayMask	Days selected values. 1 = Monday, 2 = Tuesday, 4 = Wednesday, 8 = Thursday, 16 = Friday, 32 = Saturday, 64 = Sunday, 128 = Day. Default=0. This is used with Weekly and Monthly Type recurrence. Min=1, Max=255, Default=4

uint DayMask: Days selected values. 1 = Monday, 2 = Tuesday, 4 = Wednesday, 8 = Thursday, 16 = Friday, 32 = Saturday, 64 = Sunday, 128 = Day. Default=0. This is used with

	Weekly and Monthly Type recurrence. Min=1, Max=255, Default=4
date	Comma delimited days of the month. Numbers from 1-31 are the only valid values. This is used with Monthly Date recurrence. Default="" Max Length=83

uint[] Date: Days of the month. Numbers from 1-31 are the only valid values. This is used with Monthly Date recurrence. Default="" Max Length=83

monthlyOffset	When to happen monthly. 0 = None, 1 = First, 2 = Second, 3 = Third, 4 = Forth, 5 = Last. This is used with Monthly Type recurrence. Min=0, Max=5, Default=3

uint MonthlyOffset: When to happen monthly. 0 = None, 1 = First, 2 = Second, 3 = Third, 4 = Forth, 5 = Last. This is used with Monthly Type recurrence. Min=0, Max=5, Default=3

esdType	Esd to use. Valid values are SCCM or WSUS. If no value, a custom script is run on the client. Default=null Max Length=50

string EsdType: Esd to use. Valid values are SCCM or WSUS. If no value, a custom script is run on the client. Default="" Max Length=50

preUpdateScript	Script file to run before the update starts. Default=null Max Length=255

string PreUpdateScript: Script file to run before the update starts. Default="" Max Length=255

preVmScript	Script file to run before the VM is loaded. Default=null Max Length=255

string PreVmScript: Script file to run before the VM is loaded. Default="" Max Length=255

postUpdateScript	Script file to run after the update finishes. Default=null Max Length=255

string PostUpdateScript: Script file to run after the update finishes. Default="" Max Length=255

postVmScript	Script file to run after the VM is unloaded. Default=null Max Length=255

string PostVmScript: Script file to run after the VM is unloaded. Default="" Max Length=255

domain	Domain to add the Disk Update Device(s) to. If not included, the first Domain Controller found on the Server is used. Default=null Max Length=255

string Domain: Domain to add the Disk Update Device(s) to. If not included, the first Domain Controller found on the Server is used. Default="" Max Length=255

organizationUnit	Organizational Unit to add the Disk Update Device(s) to. This parameter is optional. If

it is not specified, the device is added to the built in Computers container. Child OU's should be delimited with forward slashes, e.g. "ParentOU/ChildOU". Special characters in an OU name, such as '"', '#', '+', ',', ';', '>', '=', must be escaped with a backslash. For example, an OU called "commaIn,TheMiddle" must be specified as "commaIn\,TheMiddle". The old syntax of delimiting child OU's with a comma is still supported, but deprecated. Note that in this case, the child OU comes first, e.g. "ChildOU,ParentOU". Default=null Max Length=255

string OrganizationUnit: Organizational Unit to add the Disk Update Device(s) to. This parameter is optional. If it is not specified, the device is added to the built in Computers container. Child OU's should be delimited with forward slashes, e.g. "ParentOU/ChildOU". Special characters in an OU name, such as '"', '#', '+', ',', ';', '>', '=', must be escaped with a backslash. For example, an OU called "commaIn,TheMiddle" must be specified as "commaIn\,TheMiddle". The old syntax of delimiting child OU's with a comma is still supported, but deprecated. Note that in this case, the child OU comes first, e.g. "ChildOU,ParentOU". Default="" Max Length=255

postUpdateApprove   Access to place the version in after the update has occurred. 0 = Production, 1 = Test, 2 = Maintenance. Min=0, Max=2, Default=0

uint PostUpdateApprove: Access to place the version in after the update has occurred. 0 = Production, 1 = Test, 2 = Maintenance. Min=0, Max=2, Default=0

Optional

SwitchParameter Confirm: The impact of this operation is "low". If -Confirm is specified, the operation will be confirmed. $ConfirmPreference can be set to "low" to have confirmation without the Confirm parameter.

resultId   If successful, the GUID of the record added.

PvsUpdateTask: If successful, the new PvsUpdateTask object is returned.

Mcli-Add VirtualHostingPool

Add a new Virtual Hosting Pool for the Site.

New-PvsVirtualHostingPool

Add a new Virtual Hosting Pool to a Site.

-r   Record to add. All record fields that do not have a Default are required.

All parameters that do not have a Default are required, unless only a
few of a group are required.

virtualHostingPoolName   Name of the Virtual Hosting Pool. It is
unique within the Site. Max Length=50

string Name or VirtualHostingPoolName: Name of the Virtual Hosting
Pool. It is unique within the Site. Max Length=50

siteId   GUID of the Site that this Virtual Hosting
Pool is a member of. It is not used with
siteName. Length=36

Guid SiteId: GUID of the Site that this Virtual Hosting Pool is a
member of. It is not used with SiteName.

siteName   Name of the Site that this Virtual Hosting
Pool is a member of. It is not used with
siteId.

string SiteName: Name of the Site that this Virtual Hosting Pool is a
member of. It is not used with SiteId.

type   Type of the Virtual Hosting Pool. 0 = Citrix
XenServer, 1 = Microsoft SCVMM/Hyper-V, 2 =
VMWare vSphere/ESX. Min=0, Max=3, Default=0

uint Type: Type of the Virtual Hosting Pool. 0 = Citrix XenServer, 1
= Microsoft SCVMM/Hyper-V, 2 = VMWare vSphere/ESX.
Min=0, Max=3, Default=0

description   User description. Default="" Max Length=250

string Description: User description. Default="" Max Length=250

server   Name or IP of the Host Server. Max
Length=255

string Server: Name or IP of the Host Server. Max Length=255

port   Port of the Host Server. Min=80, Max=65534,
Default=80

uint Port: Port of the Host Server. Min=80, Max=65534, Default=80

datacenter   Datacenter of the Virtual Hosting Pool.
Default="" Max Length=250

string Datacenter: Datacenter of the Virtual Hosting Pool. Default=""
Max Length=250

updateLimit   Number of updates at the same time. Min=2,
Max=1000, Default=1000

uint UpdateLimit: Number of updates at the same time. Min=2,
Max=1000, Default=1000

updateTimeout   Timeout for updates. Min=2, Max=240,
Default=60

uint UpdateTimeout: Timeout for updates. Min=2, Max=240, Default=60

shutdownTimeout   Timeout for shutdown. Min=2, Max=30,
Default=10

uint ShutdownTimeout: Timeout for shutdown. Min=2, Max=30, Default=10

userName   Name to use when logging into the Server.

   string UserName: Name to use when logging into the Server.

password   Password to use when logging into the
                                   Server.

  string Password: Password to use when logging into the Server.

xdHostingUnitUuid   UUID of XenDesktop Hosting Unit Default=null
                                   Length=36

  Guid XdHostingUnitUuid: UUID of XenDesktop Hosting Unit
                 Default=00000000-0000-0000-0000-000000000000

prepopulateEnabled   Enable prepopulate when set to 1 Default=0

  SwitchParameter PrepopulateEnabled: Enable prepopulate when set to
                 true Default=false

xsPvsSiteUuid   UUID of XenServer PVS_site Default=null
                                   Length=36

  Guid XsPvsSiteUuid: UUID of XenServer PVS_site Default=00000000-
                 0000-0000-0000-000000000000

platformVersion   Hypervisor Host Version Default=null Max
                                   Length=250

  string PlatformVersion: Hypervisor Host Version Default="" Max
                 Length=250

xdHcHypervisorConnectionName Hypervisor Connection Name for HCL
                                 Connection Details object Default=null Max
                                 Length=250

  string XdHcHypervisorConnectionName: Hypervisor Connection Name for
                 HCL Connection Details object Default="" Max
                 Length=250

xdHcHypervisorConnectionUid Hypervisor Connection Uid for HCL
                                 Connection Details object Default=null Max
                                 Length=250

  string XdHcHypervisorConnectionUid: Hypervisor Connection Uid for HCL
                 Connection Details object Default="" Max Length=250

xdHcRevision   Revision for HCL Connection Details object
                                 Default=null Max Length=250

  string XdHcRevision: Revision for HCL Connection Details object
                 Default="" Max Length=250

xdHcCustomProperties   Custom Properties for HCL Connection Details
                                 object Default=null Max Length=250

  string XdHcCustomProperties: Custom Properties for HCL Connection
                 Details object Default="" Max Length=250

xdHcSslThumbprints   Ssl Thumbprints for HCL Connection Details
                                 object Default=null Max Length=250

  string XdHcSslThumbprints: Ssl Thumbprints for HCL Connection Details
                 object Default="" Max Length=250

disableHostXsProxy   True to disable PVS-Accelerator Default=null

  string DisableHostXsProxy: True to disable PVS-Accelerator Default=""

Optional

resultId   If successful, the GUID of the record added.

PvsVirtualHostingPool: If successful, the new PvsVirtualHostingPool object is returned.

Mcli-Delete

```
Mcli-Delete commandName[ -p name=value[, name2=value2]]
```

Mcli-Delete AuthGroup

Delete an AuthGroup Active Directory or Windows Group name.

Remove-PvsAuthGroup

Remove one or more AuthGroup Active Directory or Windows Group names.

  -p   Parameters needed for this Delete.

 One of these required

  authGroupId   GUID of the AuthGroup to Delete.

    Guid[] Guid or AuthGroupId: GUID of the AuthGroup to Delete.

  authGroupName   Name of the AuthGroup to Delete.

    string[] Name or AuthGroupName: Name of the AuthGroup to Delete.

 Optional

  force   When set to 1, the AuthGroup will be Deleted
                              even if being used, otherwise an error is
                              returned if being used.

    SwitchParameter Force: If -Force is specified, the AuthGroup will be
                   Deleted even if being used, otherwise an error is
                   returned if being used.

  Instead of a parameter that matches one of the members listed

    PvsObject[] Object: PvsObjects with the members below can be used as
                   the Object parameter or from a pileline:

      AuthGroupId

   Optional

    SwitchParameter Confirm: The impact of this operation is "medium". If
                   -Confirm is specified, the operation will be
                   confirmed. $ConfirmPreference can be set to "medium"
                   or "low" to have confirmation without the Confirm
                   parameter.

Mcli-Delete Collection

Delete a Collection.

Remove-PvsCollection

Remove one or more Collections.

```
     -p                           Parameters needed for this Delete.
   This required
    collectionId                 GUID of the Collection to Delete.
      Guid[] Guid or CollectionId: GUID of the Collection to Delete.
   or this required & resolution
    collectionName               Name of the Collection to Delete.
      string[] Name or CollectionName: Name of the Collection to Delete.
   One of these resolutions when needed
    siteId                       GUID of the Site.
      Guid[] SiteId: GUID of the Site.
    siteName                     Name of the Site.
      string[] SiteName: Name of the Site.
   Instead of a parameter that matches one of the members listed
      PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:
       CollectionId
   Optional
      SwitchParameter Confirm: The impact of this operation is "medium". If
                    -Confirm is specified, the operation will be
                    confirmed. $ConfirmPreference can be set to "medium"
                    or "low" to have confirmation without the Confirm
                    parameter.
```

Mcli-Delete Device

```
   Delete one or more Devices.
```

Remove-PvsDevice

```
   Remove one or more Devices.
     -p                           Parameters needed for this Delete.
    One of these required
     deviceId                     GUID of the Device to Delete.
       Guid[] Guid or DeviceId: GUID of the Device to Delete.
     deviceName                   Name of the Device to Delete.
       string[] Name or DeviceName: Name of the Device to Delete.
     deviceMac                    MAC of the Device to Delete.
       PvsPhysicalAddress[] DeviceMac: MAC of the Device to Delete.
     collectionId                 GUID of the Collection to delete all
                                    Devices.
       Guid[] CollectionId: GUID of the Collection to delete all Devices.
    or this required & resolution
```

```
        collectionName              Name of the Collection to delete all
                                     Devices.

      string[] CollectionName: Name of the Collection to delete all
                     Devices.

    One of these resolutions when needed

     siteId                          GUID of the Site.

       Guid[] SiteId: GUID of the Site.

     siteName                        Name of the Site.

       string[] SiteName: Name of the Site.

    Instead of a parameter that matches one of the members listed

      PvsObject[] Object: PvsObjects with the members below can be used as
                     the Object parameter or from a pileline:

       DeviceId or CollectionId

    Optional

      SwitchParameter Confirm: The impact of this operation is "medium". If
                     -Confirm is specified, the operation will be
                     confirmed. $ConfirmPreference can be set to "medium"
                     or "low" to have confirmation without the Confirm
                     parameter.
```

Mcli-Delete DeviceDiskCacheFile

```
    Delete the Disk cache files for a Device.
```

Remove-PvsDeviceDiskCacheFile

```
    Remove one or more Disk cache files for Devices.

     -p                            Parameters needed for this Delete.

      One of these required

      deviceId                     GUID of the Device to Delete Disk cache
                                    files.

        Guid[] Guid or DeviceId: GUID of the Device to Delete Disk cache
                      files.

      deviceName                   Name of the Device to Delete Disk cache
                                    files.

        string[] Name or DeviceName: Name of the Device to Delete Disk cache
                      files.

      deviceMac                    MAC of the Device to Delete Disk cache
                                    files.

       PvsPhysicalAddress[] DeviceMac: MAC of the Device to Delete Disk
                     cache files.

    This required

      diskLocatorId                GUID of the Disk Locator to Delete Disk
                                    cache files.

        Guid[] DiskLocatorId: GUID of the Disk Locator to Delete Disk cache
                      files.
```

or this required & resolution

  diskLocatorName   Name of the Disk Locator File to Delete Disk
                                  cache files.

    string[] DiskLocatorName: Name of the Disk Locator File to Delete
                     Disk cache files.

One of these resolutions when needed

  siteId   GUID of the Site.

    Guid[] SiteId: GUID of the Site.

  siteName   Name of the Site.

    string[] SiteName: Name of the Site.

One of these resolutions when needed

  storeId   GUID of the Store that is needed when a
                                  diskLocatorName is used.

    Guid[] StoreId: GUID of the Store that is needed when a
                    DiskLocatorName is used.

  storeName   Name of the Store that is needed when a
                                  diskLocatorName is used.

    string[] StoreName: Name of the Store that is needed when a
                    DiskLocatorName is used.

Instead of a parameter that matches one of the members listed

    PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

     DeviceId or DiskLocatorId

Optional

    SwitchParameter Confirm: The impact of this operation is "medium". If
                    -Confirm is specified, the operation will be
                    confirmed. $ConfirmPreference can be set to "medium"
                    or "low" to have confirmation without the Confirm
                    parameter.

Mcli-Delete DiskLocator

Delete one or more Disk Locators.

Remove-PvsDiskLocator

Remove one or more Disk Locators.

  -p   Parameters needed for this Delete.

This required

  diskLocatorId   GUID of the Disk Locator to Delete.

    Guid[] Guid or DiskLocatorId: GUID of the Disk Locator to Delete.

or one of these required & resolutions

  diskLocatorName   Name of the Disk Locator File to Delete.

    string[] Name or DiskLocatorName: Name of the Disk Locator File to
                    Delete.

```
    storeId                 GUID of the Store to delete all
                            DiskLocators.

    Guid[] StoreId: GUID of the Store to delete all DiskLocators.

    storeName               Name of the Store to delete all
                            DiskLocators.

    string[] StoreName: Name of the Store to delete all DiskLocators.

  Optional

    deleteDiskFile          1 if the Disk File should be deleted, 0
                            otherwise. Default=0

    SwitchParameter DeleteDiskFile: If -DiskFile is specified, the Disk
                    File will be deleted.

  One of these resolutions when needed

    siteId                  GUID of the Site.

    Guid[] SiteId: GUID of the Site.

    siteName                Name of the Site.

    string[] SiteName: Name of the Site.

  One of these resolutions when needed

    storeId                 GUID of the Store that is needed when a
                            diskLocatorName is used.

    Guid[] StoreId: GUID of the Store to delete all DiskLocators.

    storeName               Name of the Store that is needed when a
                            diskLocatorName is used.

    string[] StoreName: Name of the Store to delete all DiskLocators.

  Instead of a parameter that matches one of the members listed

    PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

      DiskLocatorId or StoreId

  Optional

    SwitchParameter Confirm: The impact of this operation is "medium". If
                    -Confirm is specified, the operation will be
                    confirmed. $ConfirmPreference can be set to "medium"
                    or "low" to have confirmation without the Confirm
                    parameter.
```

Mcli-Delete DiskUpdateDevice

```
    Delete a Disk Update Device.
```

Remove-PvsDiskUpdateDevice

```
    Remove one or more Disk Update Devices.

    -p                      Parameters needed for this Delete.

  One of these required

    deviceId                GUID of the Disk Update Device to Delete.
```

        deviceName   Name of the Disk Update Device to Delete.

        deviceMac   MAC of the Disk Update Device to Delete.

    Instead of a parameter that matches one of the members listed

                        the Object parameter or from a pileline:

          DeviceId

    Optional

Mcli-Delete DiskVersion

    Remove the latest Disk version or no longer needed version if no Devices
    are currently booted from that version.

Remove-PvsDiskVersion

    -p   Parameters needed for this Delete.

    This required

    diskLocatorId   GUID of the Disk Locator to Delete the
                                    Version from.

        Guid[] Guid or DiskLocatorId: GUID of the Disk Locator to Delete the
                        Version from.

    or this required & resolution

    diskLocatorName   Name of the Disk Locator File to Delete the
                                    Version from.

        string[] Name or DiskLocatorName: Name of the Disk Locator File to
                        Delete the Version from.

    Optional

    version   Specifies the version that should be
                                    deleted. Used when deleting versions that
                                    are no longer needed because of a Merge.

        uint Version: Specifies the version that should be deleted. Used when
                        deleting versions that are no longer needed because
                        of a Merge.

    One of these resolutions when needed

```
          siteId                     GUID of the Site.

           Guid[] SiteId: GUID of the Site.

          siteName                   Name of the Site.

           string[] SiteName: Name of the Site.

      One of these resolutions when needed

          storeId                    GUID of the Store that is needed when a
                                      diskLocatorName is used.

           Guid[] StoreId: GUID of the Store that is needed when a
                      DiskLocatorName is used.

          storeName                  Name of the Store that is needed when a
                                      diskLocatorName is used.

           string[] StoreName: Name of the Store that is needed when a
                      DiskLocatorName is used.

      Instead of a parameter that matches one of the members listed

           PvsObject[] Object: PvsObjects with the members below can be used as
                      the Object parameter or from a pileline:

            (DiskLocatorId and Version) or DiskLocatorId

       Optional

           SwitchParameter Confirm: The impact of this operation is "medium". If
                      -Confirm is specified, the operation will be
                      confirmed. $ConfirmPreference can be set to "medium"
                      or "low" to have confirmation without the Confirm
                      parameter.
```

Mcli-Delete FarmView

```
    Delete a View from the Farm.
```

Remove-PvsFarmView

```
   Remove one or more Views from the Farm.

     -p                          Parameters needed for this Delete.

      One of these required

       farmViewId                GUID of the Farm View to Delete.

         Guid[] Guid or FarmViewId: GUID of the Farm View to Delete.

       farmViewName              Name of the Farm View to Delete.

         string[] Name or FarmViewName: Name of the Farm View to Delete.

       Instead of a parameter that matches one of the members listed

         PvsObject[] Object: PvsObjects with the members below can be used as
                     the Object parameter or from a pileline:

           FarmViewId

       Optional

         SwitchParameter Confirm: The impact of this operation is "medium". If
                     -Confirm is specified, the operation will be
                     confirmed. $ConfirmPreference can be set to "medium"
```

Mcli-Delete Server

Delete a Server.

Remove-PvsServer

Remove one or more Servers.

```
-p                         Parameters needed for this Delete.

 One of these required

  serverId                 GUID of the Server to Delete.

   Guid[] Guid or ServerId: GUID of the Server to Delete.

  serverName               Name of the Server to Delete.

   string[] Name or ServerName: Name of the Server to Delete.

  Instead of a parameter that matches one of the members listed

   PvsObject[] Object: PvsObjects with the members below can be used as
                   the Object parameter or from a pileline:

     ServerId

  Optional

   SwitchParameter Confirm: The impact of this operation is "medium". If
                   -Confirm is specified, the operation will be
                   confirmed. $ConfirmPreference can be set to "medium"
                   or "low" to have confirmation without the Confirm
                   parameter.
```

Mcli-Delete ServerStore

Delete the connection from a Server to a Store.

Remove-PvsServerStore

Remove the connection from Servers to Stores.

```
-p                         Parameters needed for this Delete.

 One of these required

  serverId                 GUID of a Server that uses the path to get
                            to the Store.

   Guid[] Guid or ServerId: GUID of a Server that uses the path to get
                   to the Store.

  serverName               Name of a Server that uses the path to get
                            to the Store.

   string[] Name or ServerName: Name of a Server that uses the path to
                   get to the Store.

 One of these required

  storeId                  GUID of the Store.

   Guid[] StoreId: GUID of the Store.
```

```
        storeName                    Name of the Store.

          string[] StoreName: Name of the Store.

        Instead of a parameter that matches one of the members listed

          PvsObject[] Object: PvsObjects with the members below can be used as
                          the Object parameter or from a pileline:

            ServerId or StoreId

        Optional

          SwitchParameter Confirm: The impact of this operation is "medium". If
                          -Confirm is specified, the operation will be
                          confirmed. $ConfirmPreference can be set to "medium"
                          or "low" to have confirmation without the Confirm
                          parameter.
```

Mcli-Delete Site

```
   Delete a Site.
```

Remove-PvsSite

```
   Remove one or more Sites.

     -p                            Parameters needed for this Delete.

      One of these required

      siteId                       GUID of the Site to Delete.

        Guid[] Guid or SiteId: GUID of the Site to Delete.

      siteName                     Name of the Site to Delete.

        string[] Name or SiteName: Name of the Site to Delete.

      Instead of a parameter that matches one of the members listed

        PvsObject[] Object: PvsObjects with the members below can be used as
                        the Object parameter or from a pileline:

          SiteId

      Optional

        SwitchParameter Confirm: The impact of this operation is "medium". If
                        -Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "medium"
                        or "low" to have confirmation without the Confirm
                        parameter.
```

Mcli-Delete SiteView

```
   Delete a View from a Site.
```

Remove-PvsSiteView

```
   Remove one or more Views from Sites.

     -p                            Parameters needed for this Delete.

      This required

      siteViewId                   GUID of the Site View to Delete.
```

Guid[] Guid or SiteViewId: GUID of the Site View to Delete.
    or this required & resolution
      siteViewName   Name of the Site View to Delete.
        string[] Name or SiteViewName: Name of the Site View to Delete.
    One of these resolutions when needed
      siteId   GUID of the Site.
        Guid[] SiteId: GUID of the Site.
      siteName   Name of the Site.
        string[] SiteName: Name of the Site.
    Instead of a parameter that matches one of the members listed
        PvsObject[] Object: PvsObjects with the members below can be used as
                        the Object parameter or from a pileline:

          SiteViewId

      Optional

        SwitchParameter Confirm: The impact of this operation is "medium". If
                        -Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "medium"
                        or "low" to have confirmation without the Confirm
                        parameter.

Mcli-Delete Store

    Delete a Store.

Remove-PvsStore

    Remove one or more Stores.

      -p   Parameters needed for this Delete.
     One of these required
      storeId   GUID of the Store to Delete.
        Guid[] Guid or StoreId: GUID of the Store to Delete.
      storeName   Name of the Store to Delete.
        string[] Name or StoreName: Name of the Store to Delete.
    Instead of a parameter that matches one of the members listed
        PvsObject[] Object: PvsObjects with the members below can be used as
                      the Object parameter or from a pileline:

          StoreId

      Optional

        SwitchParameter Confirm: The impact of this operation is "medium". If
                        -Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "medium"
                        or "low" to have confirmation without the Confirm
                        parameter.

Mcli-Delete UpdateTask

Delete an Update Task from a Site.

Remove-PvsUpdateTask

Remove one or more Update Tasks from Sites.

  -p   Parameters needed for this Delete.

  This required

   updateTaskId   GUID of the Update Task to Delete.

     Guid[] Guid or UpdateTaskId: GUID of the Update Task to Delete.

  or this required & resolution

   updateTaskName   Name of the Update Task to Delete.

     string[] Name or UpdateTaskName: Name of the Update Task to Delete.

  One of these resolutions when needed

   siteId   GUID of the Site.

     Guid[] SiteId: GUID of the Site.

   siteName   Name of the Site.

     string[] SiteName: Name of the Site.

  Instead of a parameter that matches one of the members listed

   PvsObject[] Object: PvsObjects with the members below can be used as the Object parameter or from a pileline:

    UpdateTaskId

  Optional

   SwitchParameter Confirm: The impact of this operation is "medium". If -Confirm is specified, the operation will be confirmed. $ConfirmPreference can be set to "medium" or "low" to have confirmation without the Confirm parameter.

Mcli-Delete VirtualHostingPool

Delete a Virtual Hosting Pool from a Site.

Remove-PvsVirtualHostingPool

Remove one or more Virtual Hosting Pools from Sites.

  -p   Parameters needed for this Delete.

  This required

   virtualHostingPoolId   GUID of the Virtual Hosting Pool to Delete.

     Guid[] Guid or VirtualHostingPoolId: GUID of the Virtual Hosting Pool to Delete.

  or this required & resolution

   virtualHostingPoolName   Name of the Virtual Hosting Pool to Delete.

     string[] Name or VirtualHostingPoolName: Name of the Virtual Hosting Pool to Delete.

```
    One of these resolutions when needed

     siteId                          GUID of the Site.

       Guid[] SiteId: GUID of the Site.

     siteName                        Name of the Site.

       string[] SiteName: Name of the Site.

     Instead of a parameter that matches one of the members listed

       PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

        VirtualHostingPoolId

     Optional

       SwitchParameter Confirm: The impact of this operation is "medium". If
                    -Confirm is specified, the operation will be
                    confirmed. $ConfirmPreference can be set to "medium"
                    or "low" to have confirmation without the Confirm
                    parameter.
```

Mcli-Get

```
 Mcli-Get commandName[ -p name=value[, name2=value2]][ -s field[-]][ -f
    field[, field2]]
```

Mcli-Get AuditActionParameters

```
    Get the Parameters of an Audit Action.
```

Get-PvsAuditActionParameter

```
  Get the Parameters of an Audit Action.

   -p                              Parameters needed for the Get.

    This required

     auditActionId                 GUID of the Audit Action to Get Parameters
                                    for.

       Guid[] AuditActionId: GUID of the Audit Action to Get Parameters for.

     Instead of a parameter that matches one of the members listed

       PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

        AuditActionId

   -f                              List of fields to retrieve, if not included,
                                    all fields are retrieved.

    If only selected fields are needed, pass them in the Fields parameter
                 as a string array.

     auditParameterName            Name of the parameter. Max Length=50

       Name or AuditParameterName: Name of the parameter. Max Length=50

     value                         Value of the parameter. Max Length=1000

       Value: Value of the parameter. Max Length=1000
```

```
    records                         If successful, records retrieved.
```

  PvsAuditActionParameter[]: If successful, the PvsAuditActionParameter
                     object(s) are returned.

Mcli-Get AuditActionProperties

```
    Get the Properties of an Audit Action.
```

Get-PvsAuditActionProperty

  Get the Properties of an Audit Action.

```
    -p                              Parameters needed for the Get.
```

   This required

```
     auditActionId               GUID of the Audit Action to Get Properties
                                 for.
```

     Guid[] AuditActionId: GUID of the Audit Action to Get Properties for.

```
     Instead of a parameter that matches one of the members listed
```

     PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

       AuditActionId

```
    -f                              List of fields to retrieve, if not included,
                                    all fields are retrieved.
```

    If only selected fields are needed, pass them in the Fields parameter
                    as a string array.

```
     auditPropertyName           Name of the property. Max Length=50
```

     Name or AuditPropertyName: Name of the property. Max Length=50

```
     oldValue                    Previous value of the Property. Default=null
                                 Max Length=1000
```

     OldValue: Previous value of the Property. Default="" Max Length=1000

```
     newValue                    New value of the Property. Default=null Max
                                 Length=1000
```

     NewValue: New value of the Property. Default="" Max Length=1000

```
    records                         If successful, records retrieved.
```

  PvsAuditActionProperty[]: If successful, the PvsAuditActionProperty
                     object(s) are returned.

Mcli-Get AuditActionSibling

```
    Get the Sibling of an Audit Action. It is the 2nd object involved with the
    action.
```

Get-PvsAuditActionSibling

  Get the Sibling of an Audit Action. It is the 2nd object involved with the
    action.

```
    -p                              Parameters needed for the Get.
```

   This required

```
    auditActionId              GUID of the Audit Action to Get Sibling for.

      Guid[] Guid or AuditActionId: GUID of the Audit Action to Get Sibling
                     for.

    Instead of a parameter that matches one of the members listed

      PvsObject[] Object: PvsObjects with the members below can be used as
                     the Object parameter or from a pileline:

        AuditActionId

-f                             List of fields to retrieve, if not included,
                               all fields are retrieved.

  If only selected fields are needed, pass them in the Fields parameter
                     as a string array.

    auditActionId              GUID of the action. Length=36

      Guid or AuditActionId: GUID of the action.

    type                       Type of object that action was performed on.
                               Values are: 1 (AuthGroup), 2 (Collection), 3
                               (Device), 4 (Disk), 5 (DiskLocator), 6
                               (Farm), 7 (FarmView), 8 (Server), 9 (Site),
                               10 (SiteView), 11 (Store), 12 (System), and
                               13 (UserGroup)

      Type: Type of object that action was performed on. Values are: 1
                     (AuthGroup), 2 (Collection), 3 (Device), 4 (Disk), 5
                     (DiskLocator), 6 (Farm), 7 (FarmView), 8 (Server), 9
                     (Site), 10 (SiteView), 11 (Store), 12 (System), and
                     13 (UserGroup)

    objectId                   GUID of the object of the action. Length=36

      ObjectId: GUID of the object of the action.

    objectName                 Name of the object of the action. Max
                               Length=1000

      ObjectName: Name of the object of the action. Max Length=1000

    path                       Path of the object of the action. An example
                               is Site\Collection for a Device.
                               Default=null Max Length=101

      Path: Path of the object of the action. An example is Site\Collection
                     for a Device. Default="" Max Length=101

    siteId                     GUID of the Site for the object of the
                               action. Empty when not valid. Default=null
                               Length=36

      SiteId: GUID of the Site for the object of the action. 00000000-0000-
                     0000-0000-000000000000 when not valid.
                     Default=00000000-0000-0000-0000-000000000000

    subId                      GUID of the Collection or Store of the
                               action. Empty when not valid. Default=null
                               Length=36

      SubId: GUID of the Collection or Store of the action. 00000000-0000-
                     0000-0000-000000000000 when not valid.
                     Default=00000000-0000-0000-0000-000000000000
```

records   If successful, records retrieved.

Mcli-Get AuditTrail

Get the Audit Trail actions for a Farm, Site, Server, DiskLocator,
   Collection, Device, User Group, Site View, Farm View or Store. All Audit
   Trail actions are returned if no parameters are passed. The result can be
   filtered by parent, user\domain and date range.

Get-PvsAuditTrail

Get the Audit Trail actions for a Farm, Site, Server, DiskLocator,
   Collection, Device, User Group, Site View, Farm View or Store. All Audit
   Trail actions are returned if no parameters are passed. The result can be
   filtered by parent, user\domain and date range.

-p   Parameters needed for the Get.

   One of these optional

   auditActionId   GUID of the Audit Action to Get.

      Guid[] Guid or AuditActionId: GUID of the Audit Action to Get.

   parentId   Parent auditActionId of the records to
                                 retrieve. If no parameters are included,
                                 only records with no parent are returned.

      Guid[] ParentId: Parent AuditActionId of the records to retrieve. If
                      no parameters are included, only records with no
                      parent are returned.

   rootId   Root auditActionId of the records to
                                 retrieve. All of the actions caused by the
                                 root action are returned. If no parameters
                                 are included, only records with no root are
                                 returned.

      Guid[] RootId: Root AuditActionId of the records to retrieve. All of
                      the actions caused by the root action are returned.
                      If no parameters are included, only records with no
                      root are returned.

   siteId   GUID of the Site to get the Audit Trail for.

      Guid[] SiteId: GUID of the Site to get the Audit Trail for.

   siteName   Name of the Site to get the Audit Trail for.

      string[] SiteName: Name of the Site to get the Audit Trail for.

   collectionId   GUID of the Collection to get the Audit
                                 Trail for.

      Guid[] CollectionId: GUID of the Collection to get the Audit Trail
                      for.

   siteViewId   GUID of the Site View to get the Audit Trail
                                 for.

      Guid[] SiteViewId: GUID of the Site View to get the Audit Trail for.

farmViewId	GUID of the Farm View to get the Audit Trail for.

Guid[] FarmViewId: GUID of the Farm View to get the Audit Trail for.

farmViewName	Name of the Farm View to get the Audit Trail for.

string[] FarmViewName: Name of the Farm View to get the Audit Trail for.

serverId	GUID of the Server to get the Audit Trail for.

Guid[] ServerId: GUID of the Server to get the Audit Trail for.

serverName	Name of the Server to get the Audit Trail for.

string[] ServerName: Name of the Server to get the Audit Trail for.

deviceId	GUID of the Device to get the Audit Trail for.

Guid[] DeviceId: GUID of the Device to get the Audit Trail for.

deviceName	Name of the Device to get the Audit Trail for.

string[] DeviceName: Name of the Device to get the Audit Trail for.

deviceMac	MAC of the Device to get the Audit Trail for.

PvsPhysicalAddress[] DeviceMac: MAC of the Device to get the Audit Trail for.

storeId	GUID of the Store to get the Audit Trail for.

Guid[] StoreId: GUID of the Store to get the Audit Trail for.

storeName	Name of the Store to get the Audit Trail for.

string[] StoreName: Name of the Store to get the Audit Trail for.

diskLocatorId	GUID of the DiskLocator to get the Audit Trail for.

Guid[] DiskLocatorId: GUID of the DiskLocator to get the Audit Trail for.

or one of these optional & resolutions

collectionName	Name of the Collection to get the Audit Trail for.

string[] CollectionName: Name of the Collection to get the Audit Trail for.

siteViewName	Name of the Site View to get the Audit Trail for.

string[] SiteViewName: Name of the Site View to get the Audit Trail for.

diskLocatorName   Name of the DiskLocator to get the Audit
                                Trail for.

  string[] DiskLocatorName: Name of the DiskLocator to get the Audit
                Trail for.

Optional

userName   User that performed the action.

  string[] UserName: User that performed the action.

domain   Domain of the user that performed the
                                action.

  string[] Domain: Domain of the user that performed the action.

beginDate   Date of the first actions to get. If not
                                included, all actions until the endDate are
                                returned. If neither this or the endDate are
                                included, then only actions that occurred in
                                the last week are returned. Format is:
                                yyyy/mm/dd

  DateTime BeginDate: Date/Time of the first actions to get. If not
                included, all actions until the endDate are
                returned. If neither this or the endDate are
                included, then only actions that occurred in the
                last week are returned. Uses only the date, hour and
                minute. If connected to a Server version previous to
                7.7, only the date is used.

endDate   Date of the last actions to get. If not
                                included, all actions from the beginDate
                                until now are returned. If neither this or
                                the beginDate are included, then only
                                actions that occurred in the last week are
                                returned. Format is: yyyy/mm/dd

  DateTime EndDate: Date/Time of the last actions to get. If not
                included, all actions from the beginDate until now
                are returned. If neither this or the beginDate are
                included, then only actions that occurred in the
                last week are returned. Uses only the date, hour and
                minute. If connected to a Server version previous to
                7.7, only the date is used.

type   Comma delimited list of types to get. Values
                                are listed in the type field description.

  uint[] Type: Types to get. Values are: 0 (Many), 1 (AuthGroup), 2
              (Collection), 3 (Device), 4 (Disk), 5 (DiskLocator),
              6 (Farm), 7 (FarmView), 8 (Server), 9 (Site), 10
              (SiteView), 11 (Store), 12 (System), and 13
              (UserGroup)

action   Comma delimited list of actions to get.
                                Values are listed in the action field
                                description.

  uint[] Action: Actions to get. Values are: 1 (AddAuthGroup), 2
              (AddCollection), 3 (AddDevice), 4 (AddDiskLocator),
              5 (AddFarmView), 6 (AddServer), 7 (AddSite), 8

(AddSiteView), 9 (AddStore), 10 (AddUserGroup), 11 (AddVirtualHostingPool), 12 (AddUpdateTask), 13 (AddDiskUpdateDevice), 1001 (DeleteAuthGroup), 1002 (DeleteCollection), 1003 (DeleteDevice), 1004 (DeleteDeviceDiskCacheFile), 1005 (DeleteDiskLocator), 1006 (DeleteFarmView), 1007 (DeleteServer), 1008 (DeleteServerStore), 1009 (DeleteSite), 1010 (DeleteSiteView), 1011 (DeleteStore), 1012 (DeleteUserGroup), 1013 (DeleteVirtualHostingPool), 1014 (DeleteUpdateTask), 1015 (DeleteDiskUpdateDevice), 1016 (DeleteDiskVersion), 2001 (RunAddDeviceToDomain), 2002 (RunApplyAutoUpdate), 2003 (RunApplyIncrementalUpdate), 2004 (RunArchiveAuditTrail), 2005 (RunAssignAuthGroup), 2006 (RunAssignDevice), 2007 (RunAssignDiskLocator), 2008 (RunAssignServer), 2009 (RunWithReturnBoot), 2010 (RunCopyPasteDevice), 2011 (RunCopyPasteDisk), 2012 (RunCopyPasteServer), 2013 (RunCreateDirectory), 2014 (RunCreateDiskCancel), 2015 (RunDisableCollection), 2016 (RunDisableDevice), 2017 (RunDisableDeviceDiskLocator), 2018 (RunDisableDiskLocator), 2019 (RunDisableUserGroup), 2020 (RunDisableUserGroupDiskLocator), 2021 (RunWithReturnDisplayMessage), 2022 (RunEnableCollection), 2023 (RunEnableDevice), 2024 (RunEnableDeviceDiskLocator), 2025 (RunEnableDiskLocator), 2026 (RunEnableUserGroup), 2027 (RunEnableUserGroupDiskLocator), 2028 (RunExportOemLicenses), 2029 (RunImportDatabase), 2030 (RunImportDevices), 2031 (RunImportOemLicenses), 2032 (RunMarkDown), 2033 (RunWithReturnReboot), 2034 (RunRemoveAuthGroup), 2035 (RunRemoveDevice), 2036 (RunRemoveDeviceFromDomain), 2037 (RunRemoveDirectory), 2038 (RunRemoveDiskLocator), 2039 (RunResetDeviceForDomain), 2040 (RunResetDatabaseConnection), 2041 (RunRestartStreamingService), 2042 (RunWithReturnShutdown), 2043 (RunStartStreamingService), 2044 (RunStopStreamingService), 2045 (RunUnlockAllDisk), 2046 (RunUnlockDisk), 2047 (RunServerStoreVolumeAccess), 2048 (RunServerStoreVolumeMode), 2049 (RunMergeDisk), 2050 (RunRevertDiskVersion), 2051 (RunPromoteDiskVersion), 2052 (RunCancelDiskMaintenance), 2053 (RunActivateDevice), 2054 (RunAddDiskVersion), 2055 (RunExportDisk), 2056 (RunAssignDisk), 2057 (RunRemoveDisk), 2058 (RunDiskUpdateStart), 2059 (RunDiskUpdateCancel), 2060 (RunSetOverrideVersion), 2061 (RunCancelTask), 2062 (RunClearTask), 2063 (RunForceInventory), 2064 RunUpdateBDM, 2065 (RunStartDeviceDiskTempVersionMode), 2066 (RunStopDeviceDiskTempVersionMode), 3001

(RunWithReturnCreateDisk), 3002
(RunWithReturnCreateDiskStatus), 3003
(RunWithReturnMapDisk), 3004
(RunWithReturnRebalanceDevices), 3005
(RunWithReturnCreateMaintenanceVersion), 3006
(RunWithReturnImportDisk), 4001
(RunByteArrayInputImportDevices), 4002
(RunByteArrayInputImportOemLicenses), 5001
(RunByteArrayOutputArchiveAuditTrail), 5002
(RunByteArrayOutputExportOemLicenses), 6001
(SetAuthGroup), 6002 (SetCollection), 6003
(SetDevice), 6004 (SetDisk), 6005 (SetDiskLocator),
6006 (SetFarm), 6007 (SetFarmView), 6008
(SetServer), 6009 (SetServerBiosBootstrap), 6010
(SetServerBootstrap), 6011 (SetServerStore), 6012
(SetSite), 6013 (SetSiteView), 6014 (SetStore), 6015
(SetUserGroup), 6016 SetVirtualHostingPool, 6017
SetUpdateTask, 6018 SetDiskUpdateDevice, 7001
(SetListDeviceBootstraps), 7002
(SetListDeviceBootstrapsDelete), 7003
(SetListDeviceBootstrapsAdd), 7004
(SetListDeviceCustomProperty), 7005
(SetListDeviceCustomPropertyDelete), 7006
(SetListDeviceCustomPropertyAdd), 7007
(SetListDeviceDiskPrinters), 7008
(SetListDeviceDiskPrintersDelete), 7009
(SetListDeviceDiskPrintersAdd), 7010
(SetListDevicePersonality), 7011
(SetListDevicePersonalityDelete), 7012
(SetListDevicePersonalityAdd), 7013
(SetListDiskLocatorCustomProperty), 7014
(SetListDiskLocatorCustomPropertyDelete), 7015
(SetListDiskLocatorCustomPropertyAdd), 7016
(SetListServerCustomProperty), 7017
(SetListServerCustomPropertyDelete), 7018
(SetListServerCustomPropertyAdd), 7019
(SetListUserGroupCustomProperty), 7020
(SetListUserGroupCustomPropertyDelete), and 7021
(SetListUserGroupCustomPropertyAdd)

One of these resolutions when needed

siteId   GUID of the Site.

  Guid[] SiteId: GUID of the Site to get the Audit Trail for.

siteName   Name of the Site.

  string[] SiteName: Name of the Site to get the Audit Trail for.

One of these resolutions when needed

storeId   GUID of the Store that is needed when a
                          diskLocatorName is used.

  Guid[] StoreId: GUID of the Store to get the Audit Trail for.

storeName   Name of the Store that is needed when a
                          diskLocatorName is used.

  string[] StoreName: Name of the Store to get the Audit Trail for.

Instead of a parameter that matches one of the members listed

PvsObject[] Object: PvsObjects with the members below can be used as the Object parameter or from a pileline:

AuditActionId, ParentId, RootId, SiteId, CollectionId, SiteViewId, FarmViewId, ServerId, DeviceId, StoreId or DiskLocatorId

-f   List of fields to retrieve, if not included, all fields are retrieved.

If only selected fields are needed, pass them in the Fields parameter as a string array.

auditActionId   GUID of the action. Length=36

Guid or AuditActionId: GUID of the action.

time   Date/Time the action occurred down to the millisecond. Format is: YYYY-MM-DD HH:MM:SS.SSS Default=getdate

Time: Date/Time the action occurred down to the millisecond. Has the date and time including milliseconds. Default=Empty

userName   User that performed the action. Max Length=255

UserName: User that performed the action. Max Length=255

domain   Domain of the user that performed the action. Max Length=255

Domain: Domain of the user that performed the action. Max Length=255

type   Type of object that action was performed on. Values are: 0 (Many), 1 (AuthGroup), 2 (Collection), 3 (Device), 4 (Disk), 5 (DiskLocator), 6 (Farm), 7 (FarmView), 8 (Server), 9 (Site), 10 (SiteView), 11 (Store), 12 (System), and 13 (UserGroup)

Type: Type of object that action was performed on. Values are: 0 (Many), 1 (AuthGroup), 2 (Collection), 3 (Device), 4 (Disk), 5 (DiskLocator), 6 (Farm), 7 (FarmView), 8 (Server), 9 (Site), 10 (SiteView), 11 (Store), 12 (System), and 13 (UserGroup)

action   Name of the action taken. This is a number that is converted to a string for display. Values are: 1 (AddAuthGroup), 2 (AddCollection), 3 (AddDevice), 4 (AddDiskLocator), 5 (AddFarmView), 6 (AddServer), 7 (AddSite), 8 (AddSiteView), 9 (AddStore), 10 (AddUserGroup), 11 (AddVirtualHostingPool), 12 (AddUpdateTask), 13 (AddDiskUpdateDevice), 1001 (DeleteAuthGroup), 1002 (DeleteCollection), 1003 (DeleteDevice), 1004 (DeleteDeviceDiskCacheFile), 1005 (DeleteDiskLocator), 1006 (DeleteFarmView), 1007 (DeleteServer), 1008 (DeleteServerStore), 1009 (DeleteSite), 1010

(DeleteSiteView), 1011 (DeleteStore), 1012
(DeleteUserGroup), 1013
(DeleteVirtualHostingPool), 1014
(DeleteUpdateTask), 1015
(DeleteDiskUpdateDevice), 1016
(DeleteDiskVersion), 2001
(RunAddDeviceToDomain), 2002
(RunApplyAutoUpdate), 2003
(RunApplyIncrementalUpdate), 2004
(RunArchiveAuditTrail), 2005
(RunAssignAuthGroup), 2006
(RunAssignDevice), 2007
(RunAssignDiskLocator), 2008
(RunAssignServer), 2009 (RunWithReturnBoot),
2010 (RunCopyPasteDevice), 2011
(RunCopyPasteDisk), 2012
(RunCopyPasteServer), 2013
(RunCreateDirectory), 2014
(RunCreateDiskCancel), 2015
(RunDisableCollection), 2016
(RunDisableDevice), 2017
(RunDisableDeviceDiskLocator), 2018
(RunDisableDiskLocator), 2019
(RunDisableUserGroup), 2020
(RunDisableUserGroupDiskLocator), 2021
(RunWithReturnDisplayMessage), 2022
(RunEnableCollection), 2023
(RunEnableDevice), 2024
(RunEnableDeviceDiskLocator), 2025
(RunEnableDiskLocator), 2026
(RunEnableUserGroup), 2027
(RunEnableUserGroupDiskLocator), 2028
(RunExportOemLicenses), 2029
(RunImportDatabase), 2030
(RunImportDevices), 2031
(RunImportOemLicenses), 2032 (RunMarkDown),
2033 (RunWithReturnReboot), 2034
(RunRemoveAuthGroup), 2035
(RunRemoveDevice), 2036
(RunRemoveDeviceFromDomain), 2037
(RunRemoveDirectory), 2038
(RunRemoveDiskLocator), 2039
(RunResetDeviceForDomain), 2040
(RunResetDatabaseConnection), 2041
(RunRestartStreamingService), 2042
(RunWithReturnShutdown), 2043
(RunStartStreamingService), 2044
(RunStopStreamingService), 2045
(RunUnlockAllDisk), 2046 (RunUnlockDisk),
2047 (RunServerStoreVolumeAccess), 2048
(RunServerStoreVolumeMode), 2049
(RunMergeDisk), 2050 (RunRevertDiskVersion),
2051 (RunPromoteDiskVersion), 2052
(RunCancelDiskMaintenance), 2053
(RunActivateDevice), 2054
(RunAddDiskVersion), 2055 (RunExportDisk),
2056 (RunAssignDisk), 2057 (RunRemoveDisk),

2058 (RunDiskUpdateStart), 2059 (RunDiskUpdateCancel), 2060 (RunSetOverrideVersion), 2061 (RunCancelTask), 2062 (RunClearTask), 2063 (RunForceInventory), 2064 RunUpdateBDM, 2065 (RunStartDeviceDiskTempVersionMode), 2066 (RunStopDeviceDiskTempVersionMode), 3001 (RunWithReturnCreateDisk), 3002 (RunWithReturnCreateDiskStatus), 3003 (RunWithReturnMapDisk), 3004 (RunWithReturnRebalanceDevices), 3005 (RunWithReturnCreateMaintenanceVersion), 3006 (RunWithReturnImportDisk), 4001 (RunByteArrayInputImportDevices), 4002 (RunByteArrayInputImportOemLicenses), 5001 (RunByteArrayOutputArchiveAuditTrail), 5002 (RunByteArrayOutputExportOemLicenses), 6001 (SetAuthGroup), 6002 (SetCollection), 6003 (SetDevice), 6004 (SetDisk), 6005 (SetDiskLocator), 6006 (SetFarm), 6007 (SetFarmView), 6008 (SetServer), 6009 (SetServerBiosBootstrap), 6010 (SetServerBootstrap), 6011 (SetServerStore), 6012 (SetSite), 6013 (SetSiteView), 6014 (SetStore), 6015 (SetUserGroup), 6016 SetVirtualHostingPool, 6017 SetUpdateTask, 6018 SetDiskUpdateDevice, 7001 (SetListDeviceBootstraps), 7002 (SetListDeviceBootstrapsDelete), 7003 (SetListDeviceBootstrapsAdd), 7004 (SetListDeviceCustomProperty), 7005 (SetListDeviceCustomPropertyDelete), 7006 (SetListDeviceCustomPropertyAdd), 7007 (SetListDeviceDiskPrinters), 7008 (SetListDeviceDiskPrintersDelete), 7009 (SetListDeviceDiskPrintersAdd), 7010 (SetListDevicePersonality), 7011 (SetListDevicePersonalityDelete), 7012 (SetListDevicePersonalityAdd), 7013 (SetListDiskLocatorCustomProperty), 7014 (SetListDiskLocatorCustomPropertyDelete), 7015 (SetListDiskLocatorCustomPropertyAdd), 7016 (SetListServerCustomProperty), 7017 (SetListServerCustomPropertyDelete), 7018 (SetListServerCustomPropertyAdd), 7019 (SetListUserGroupCustomProperty), 7020 (SetListUserGroupCustomPropertyDelete), and 7021 (SetListUserGroupCustomPropertyAdd)

Action: Name of the action taken. This is a number that is converted to a string for display. Values are: 1 (AddAuthGroup), 2 (AddCollection), 3 (AddDevice), 4 (AddDiskLocator), 5 (AddFarmView), 6 (AddServer), 7 (AddSite), 8 (AddSiteView), 9 (AddStore), 10 (AddUserGroup), 11 (AddVirtualHostingPool), 12 (AddUpdateTask), 13 (AddDiskUpdateDevice), 1001 (DeleteAuthGroup), 1002 (DeleteCollection), 1003

(DeleteDevice), 1004 (DeleteDeviceDiskCacheFile),
1005 (DeleteDiskLocator), 1006 (DeleteFarmView),
1007 (DeleteServer), 1008 (DeleteServerStore), 1009
(DeleteSite), 1010 (DeleteSiteView), 1011
(DeleteStore), 1012 (DeleteUserGroup), 1013
(DeleteVirtualHostingPool), 1014 (DeleteUpdateTask),
1015 (DeleteDiskUpdateDevice), 1016
(DeleteDiskVersion), 2001 (RunAddDeviceToDomain),
2002 (RunApplyAutoUpdate), 2003
(RunApplyIncrementalUpdate), 2004
(RunArchiveAuditTrail), 2005 (RunAssignAuthGroup),
2006 (RunAssignDevice), 2007 (RunAssignDiskLocator),
2008 (RunAssignServer), 2009 (RunWithReturnBoot),
2010 (RunCopyPasteDevice), 2011 (RunCopyPasteDisk),
2012 (RunCopyPasteServer), 2013
(RunCreateDirectory), 2014 (RunCreateDiskCancel),
2015 (RunDisableCollection), 2016
(RunDisableDevice), 2017
(RunDisableDeviceDiskLocator), 2018
(RunDisableDiskLocator), 2019 (RunDisableUserGroup),
2020 (RunDisableUserGroupDiskLocator), 2021
(RunWithReturnDisplayMessage), 2022
(RunEnableCollection), 2023 (RunEnableDevice), 2024
(RunEnableDeviceDiskLocator), 2025
(RunEnableDiskLocator), 2026 (RunEnableUserGroup),
2027 (RunEnableUserGroupDiskLocator), 2028
(RunExportOemLicenses), 2029 (RunImportDatabase),
2030 (RunImportDevices), 2031
(RunImportOemLicenses), 2032 (RunMarkDown), 2033
(RunWithReturnReboot), 2034 (RunRemoveAuthGroup),
2035 (RunRemoveDevice), 2036
(RunRemoveDeviceFromDomain), 2037
(RunRemoveDirectory), 2038 (RunRemoveDiskLocator),
2039 (RunResetDeviceForDomain), 2040
(RunResetDatabaseConnection), 2041
(RunRestartStreamingService), 2042
(RunWithReturnShutdown), 2043
(RunStartStreamingService), 2044
(RunStopStreamingService), 2045 (RunUnlockAllDisk),
2046 (RunUnlockDisk), 2047
(RunServerStoreVolumeAccess), 2048
(RunServerStoreVolumeMode), 2049 (RunMergeDisk),
2050 (RunRevertDiskVersion), 2051
(RunPromoteDiskVersion), 2052
(RunCancelDiskMaintenance), 2053
(RunActivateDevice), 2054 (RunAddDiskVersion), 2055
(RunExportDisk), 2056 (RunAssignDisk), 2057
(RunRemoveDisk), 2058 (RunDiskUpdateStart), 2059
(RunDiskUpdateCancel), 2060 (RunSetOverrideVersion),
2061 (RunCancelTask), 2062 (RunClearTask), 2063
(RunForceInventory), 2064 RunUpdateBDM, 2065
(RunStartDeviceDiskTempVersionMode), 2066
(RunStopDeviceDiskTempVersionMode), 3001
(RunWithReturnCreateDisk), 3002
(RunWithReturnCreateDiskStatus), 3003
(RunWithReturnMapDisk), 3004
(RunWithReturnRebalanceDevices), 3005

(RunWithReturnCreateMaintenanceVersion), 3006
(RunWithReturnImportDisk), 4001
(RunByteArrayInputImportDevices), 4002
(RunByteArrayInputImportOemLicenses), 5001
(RunByteArrayOutputArchiveAuditTrail), 5002
(RunByteArrayOutputExportOemLicenses), 6001
(SetAuthGroup), 6002 (SetCollection), 6003
(SetDevice), 6004 (SetDisk), 6005 (SetDiskLocator),
6006 (SetFarm), 6007 (SetFarmView), 6008
(SetServer), 6009 (SetServerBiosBootstrap), 6010
(SetServerBootstrap), 6011 (SetServerStore), 6012
(SetSite), 6013 (SetSiteView), 6014 (SetStore), 6015
(SetUserGroup), 6016 SetVirtualHostingPool, 6017
SetUpdateTask, 6018 SetDiskUpdateDevice, 7001
(SetListDeviceBootstraps), 7002
(SetListDeviceBootstrapsDelete), 7003
(SetListDeviceBootstrapsAdd), 7004
(SetListDeviceCustomProperty), 7005
(SetListDeviceCustomPropertyDelete), 7006
(SetListDeviceCustomPropertyAdd), 7007
(SetListDeviceDiskPrinters), 7008
(SetListDeviceDiskPrintersDelete), 7009
(SetListDeviceDiskPrintersAdd), 7010
(SetListDevicePersonality), 7011
(SetListDevicePersonalityDelete), 7012
(SetListDevicePersonalityAdd), 7013
(SetListDiskLocatorCustomProperty), 7014
(SetListDiskLocatorCustomPropertyDelete), 7015
(SetListDiskLocatorCustomPropertyAdd), 7016
(SetListServerCustomProperty), 7017
(SetListServerCustomPropertyDelete), 7018
(SetListServerCustomPropertyAdd), 7019
(SetListUserGroupCustomProperty), 7020
(SetListUserGroupCustomPropertyDelete), and 7021
(SetListUserGroupCustomPropertyAdd)

objectId   GUID of the object of the action.
                              Default=null Length=36

 ObjectId: GUID of the object of the action. Default=00000000-0000-
                0000-0000-000000000000

objectName   Name of the object of the action.
                              Default=null Max Length=1000

 ObjectName: Name of the object of the action. Default="" Max
                Length=1000

path   Path of the object of the action. An example
                              is Site\Collection for a Device.
                              Default=null Max Length=101

 Path: Path of the object of the action. An example is Site\Collection
                for a Device. Default="" Max Length=101

siteId   GUID of the Site for the object of the
                              action. Empty when not valid. Default=null
                              Length=36

subId   GUID of the Collection or Store of the action. Empty when not valid. Default=null Length=36

parentId   GUID of the parent action (one that triggered this action) if one exists. Empty when not valid. Default=null Length=36

rootId   GUID of the root action (one that triggered this group of actions) if one exists. Empty when not valid. Default=null Length=36

attachments   An or'ed value that indicates if there are any details for this action. A value of 15 indicates that there are Children, Sibling, Parameters and Properties for the action. Values are: 0 (None), 1 (Children), 2 (Sibling), 4 (Parameters), and 8 (Properties) Default=0

records   If successful, records retrieved.

PvsAuditTrail[]: If successful, the PvsAuditTrail object(s) are returned.

Mcli-Get AuthGroup

Get the fields for an AuthGroup, all AuthGroups in the system, AuthGroups with Farm, Site or Collection Authorization. All AuthGroups in the system are returned if no parameters are passed.

Get-PvsAuthGroup

Get the fields for an AuthGroup, all AuthGroups in the system, AuthGroups with Farm, Site or Collection Authorization. All AuthGroups in the system are returned if no parameters are passed.

-p   Parameters needed for the Get.

```
One of these optional

  authGroupId              GUID of the AuthGroup to Get.

    Guid[] Guid or AuthGroupId: GUID of the AuthGroup to Get.

  authGroupName            Name of the AuthGroup to Get.

    string[] Name or AuthGroupName: Name of the AuthGroup to Get.

  siteId                   GUID of the Site to Get all AuthGroups with
                           Authorization for.

    Guid[] SiteId: GUID of the Site to Get all AuthGroups with
                   Authorization for.

  siteName                 Name of the Site to Get all AuthGroups with
                           Authorization for.

    string[] SiteName: Name of the Site to Get all AuthGroups with
                   Authorization for.

  collectionId             GUID of the Collection to Get all AuthGroups
                           with Authorization for.

    Guid[] CollectionId: GUID of the Collection to Get all AuthGroups
                   with Authorization for.

or this optional & resolution

  collectionName           Name of the Collection to Get all AuthGroups
                           with Authorization for.

    string[] CollectionName: Name of the Collection to Get all AuthGroups
                   with Authorization for.

Optional

  farm                     1 if AuthGroups with Farm Authorization
                           should be returned, 0 otherwise.

    SwitchParameter Farm: If -Farm is specified, AuthGroups with Farm
                   Authorization should be returned.

One of these resolutions when needed

  siteId                   GUID of the Site.

    Guid[] SiteId: GUID of the Site to Get all AuthGroups with
                   Authorization for.

  siteName                 Name of the Site.

    string[] SiteName: Name of the Site to Get all AuthGroups with
                   Authorization for.

  Instead of a parameter that matches one of the members listed

    PvsObject[] Object: PvsObjects with the members below can be used as
                   the Object parameter or from a pileline:

      AuthGroupId, SiteId or CollectionId

-s                         Name of field to sort on. If not included,
                           the first field listed is used. Append a - to
                           the field name to get descending order.

  authGroupName            Name of the Active Directory or Windows
                           Group. Max Length=450
```

```
        description              User description. Default="" Max Length=250

   -f                            List of fields to retrieve, if not included,
                                 all fields are retrieved.

    If only selected fields are needed, pass them in the Fields parameter
                     as a string array.

      authGroupId                Read-only GUID that uniquely identifies this
                                 AuthGroup. Length=36

      Guid or AuthGroupId: Read-only GUID that uniquely identifies this
                      AuthGroup.

      authGroupName              Name of the Active Directory or Windows
                                 Group. Max Length=450

      Name or AuthGroupName: Name of the Active Directory or Windows Group.
                      Max Length=450

      description                User description. Default="" Max Length=250

       Description: User description. Default="" Max Length=250

      role                       Role of the AuthGroup for a Collection. role
                                 can only be used with collectionId or
                                 collectionName. 300 is Collection
                                 Administrator, and 400 is Collection
                                 Operator. Default=999

       Role: Role of the AuthGroup for a Collection. role can only be used
                      with CollectionId or CollectionName. 300 is
                      Collection Administrator, and 400 is Collection
                      Operator. Default=999

   records                       If successful, records retrieved.

     PvsAuthGroup[]: If successful, the PvsAuthGroup object(s) are returned.
```

Mcli-Get AuthGroupUsage

```
   Get the items that are authorized for an AuthGroup.
```

Get-PvsAuthGroupUsage

```
   Get the items that are authorized for an AuthGroup.

   -p                            Parameters needed for the Get.

    One of these required

     authGroupId                 GUID of the AuthGroup to Get all items that
                                 are authorized for it.

       Guid[] AuthGroupId: GUID of the AuthGroup to Get all items that are
                      authorized for it.

     authGroupName               Name of the AuthGroup to Get all items that
                                 are authorized for it.

       string[] Name or AuthGroupName: Name of the AuthGroup to Get all
                      items that are authorized for it.

    Instead of a parameter that matches one of the members listed

       PvsObject[] Object: PvsObjects with the members below can be used as
                      the Object parameter or from a pileline:
```

```
-f                          List of fields to retrieve, if not included,
                            all fields are retrieved.
```

If only selected fields are needed, pass them in the Fields parameter as a string array.

```
    id                      GUID of the item. The item can be a Farm,
                            Site or Collection. It will be empty for
                            Farm.
```

Guid or Id: GUID of the item. The item can be a Farm, Site or Collection. It will be 00000000-0000-0000-0000-000000000000 for Farm.

```
    name                    Name of the item. The item can be a Farm,
                            Site or Collection.
```

Name: Name of the item. The item can be a Farm, Site or Collection.

```
    role                    Role of the AuthGroup for the item. 100 is
                            Farm Administrator, 200 is Site
                            Administrator, 300 is Collection
                            Administrator, and 400 is Collection
                            Operator. Default=999
```

Role: Role of the AuthGroup for the item. 100 is Farm Administrator, 200 is Site Administrator, 300 is Collection Administrator, and 400 is Collection Operator. Default=999

```
  records                   If successful, records retrieved.
```

PvsAuthGroupUsage[]: If successful, the PvsAuthGroupUsage object(s) are returned.

Mcli-Get CeipData

```
Get the CEIP configuration
```

Get-PvsCeipData

Get the CEIP configuration

```
  -p                        Parameters needed for the Get.
```

Optional

```
    uuid                    CEIP UUID of this Farm. This is optional
                            since there is only one.
```

string[] Uuid: CEIP UUID of this Farm. This is optional since there is only one.

Instead of a parameter that matches one of the members listed

PvsObject[] Object: PvsObjects with the members below can be used as the Object parameter or from a pileline:

Uuid

```
  -f                          List of fields to retrieve, if not included,
                              all fields are retrieved.
```

enabled   1 if CEIP is enabled, otherwise 0. Min=0,
                                 Max=1

Enabled: 1 if CEIP is enabled, otherwise 0. Min=0, Max=1

uuid   CEIP UUID. Length=36

Uuid: CEIP UUID.

nextUpload   Date and time next CEIP upload is due if
                                 enabled is 1. Default=null

NextUpload: Date and time next CEIP upload is due if enabled is 1.
            Default=Empty

inProgress   1 if an upload is currently in progress,
                                 otherwise 0. Default=0

InProgress: 1 if an upload is currently in progress, otherwise 0.
            Default=0

serverId   ID of server that is currently uploading,
                                 null if inProgress is 0. Default=null
                                 Length=36

ServerId: ID of server that is currently uploading, null if
          InProgress is 0. Default=00000000-0000-0000-0000-
          000000000000

oneTimeUpload   1 to perform a one time upload. Default=0

OneTimeUpload: 1 to perform a one time upload. Default=0

records   If successful, records retrieved.

PvsCeipData[]: If successful, the PvsCeipData object(s) are returned.

Mcli-Get CisData

Get the CIS configuration

Get-PvsCisData

Get the CIS configuration

-f   List of fields to retrieve, if not included,
                                 all fields are retrieved.

If only selected fields are needed, pass them in the Fields parameter
                as a string array.

cisDataId   CIS UUID Length=36

Guid or CisDataId: CIS UUID

userName   Username used to obtain the token
                                 Default=null Max Length=255

UserName: Username used to obtain the token Default="" Max Length=255

uploadToken   Token for uploading bundles to CIS
                                 Default=null Max Length=10

UploadToken: Token for uploading bundles to CIS Default="" Max
             Length=10

```
         path                      Path where the last problem report bundle
                                   was saved Default=null Max Length=255

        Path: Path where the last problem report bundle was saved Default=""
                           Max Length=255

         password                  Password of the user required to obtain the
                                   token. This is required only by Set and Add

        Password: Password of the user required to obtain the token. This is
                        required only by Set and Add

      records                      If successful, records retrieved.

       PvsCisData[]: If successful, the PvsCisData object(s) are returned.
```

Mcli-Get Collection

```
   Get the fields for a Collection or all Collections in a Site or Farm. All
   Collections are returned if no parameters are passed.
```

Get-PvsCollection

```
   Get the fields for a Collection or all Collections in a Site or Farm. All
   Collections are returned if no parameters are passed.

      -p                           Parameters needed for the Get.

       One of these optional

        collectionId               GUID of the Collection to Get.

         Guid[] Guid or CollectionId: GUID of the Collection to Get.

        siteId                     GUID of the Site to Get all Collections for.

         Guid[] SiteId: GUID of the Site to Get all Collections for.

        siteName                   Name of the Site to Get all Collections for.

         string[] SiteName: Name of the Site to Get all Collections for.

       or this optional & resolution

        collectionName             Name of the Collection to Get.

         string[] Name or CollectionName: Name of the Collection to Get.

       One of these resolutions when needed

        siteId                     GUID of the Site.

         Guid[] SiteId: GUID of the Site to Get all Collections for.

        siteName                   Name of the Site.

         string[] SiteName: Name of the Site to Get all Collections for.

        Instead of a parameter that matches one of the members listed

         PvsObject[] Object: PvsObjects with the members below can be used as
                       the Object parameter or from a pileline:

          CollectionId or SiteId

      -s                           Name of field to sort on. If not included,
                                   the first field listed is used. Append a - to
                                   the field name to get descending order.
```

collectionName		Name of the Collection. It is unique within the Site. Max Length=50
description		User description. Default="" Max Length=250
deviceCount		Read-only count of Devices in this Collection. Default=0
deviceWithPVDCount		Read-only count of Devices with Personal vDisk in this Collection. Default=0
activeDeviceCount		Read-only count of active Devices in this Collection. Default=0
-f		List of fields to retrieve, if not included, all fields are retrieved.

If only selected fields are needed, pass them in the Fields parameter as a string array.

collectionId	Read-only GUID that uniquely identifies this Collection. Length=36

Guid or CollectionId: Read-only GUID that uniquely identifies this Collection.

collectionName	Name of the Collection. It is unique within the Site. Max Length=50

Name or CollectionName: Name of the Collection. It is unique within the Site. Max Length=50

siteId	GUID of the Site that this Collection is a member of. It is not used with siteName. Length=36

SiteId: GUID of the Site that this Collection is a member of. It is not used with SiteName.

siteName	Name of the Site that this Collection is a member of. It is not used with siteId.

SiteName: Name of the Site that this Collection is a member of. It is not used with SiteId.

description	User description. Default="" Max Length=250

Description: User description. Default="" Max Length=250

templateDeviceId	GUID of a Device in the Collection whose settings are used for initial values of new Devices. Not used with templateDeviceName. Default=null Length=36

TemplateDeviceId: GUID of a Device in the Collection whose settings are used for initial values of new Devices. Not used with templateDeviceName. Default=00000000-0000-0000-0000-000000000000

templateDeviceName	Name of a Device in the Collection whose settings are used for initial values of new Devices. Not used with templateDeviceId. Default=null

lastAutoAddDeviceNumber   The Device Number of the last Auto Added Device. Default=0

enabled   1 when Devices in the Collection can be booted, 0 otherwise. Default=1

deviceCount   Read-only count of Devices in this Collection. Default=0

deviceWithPVDCount   Read-only count of Devices with Personal vDisk in this Collection. Default=0

activeDeviceCount   Read-only count of active Devices in this Collection. Default=0

makActivateNeededCount   Read-only count of active Devices that need MAK activation in this Collection. Default=0

autoAddPrefix   The string put before the Device Number for Auto Add. Default=null ASCII computer name characters no end digit Max Length=12

autoAddSuffix   The string put after the Device Number for Auto Add. Default=null ASCII computer name characters no begin digit Max Length=12

autoAddZeroFill   1 when zeros be placed before the Device Number up to the autoAddNumberLength for Auto Add, 0 otherwise. Default=1

autoAddNumberLength   The maximum length of the Device Number for Auto Add. This length plus the autoAddPrefix length plus the autoAddSuffix length must be

less than 16. Required that
((lenautoAddPrefix+lenautoAddSuffix)+autoAdd
NumberLength)<=15. Min=3, Max=9, Default=4

AutoAddNumberLength: The maximum length of the Device Number for Auto
Add. This length plus the AutoAddPrefix length plus
the AutoAddSuffix length must be less than 16.
Required that
((lenautoAddPrefix+lenautoAddSuffix)+AutoAddNumberLe
ngth)<=15. Min=3, Max=9, Default=4

role   Read-only Role of the user for this item.
                              100 is Farm Administrator, 200 is Site
                              Administrator, 300 is Collection
                              Administrator, and 400 is Collection
                              Operator. Default=999

Role: Read-only Role of the user for this item. 100 is Farm
                   Administrator, 200 is Site Administrator, 300 is
                   Collection Administrator, and 400 is Collection
                   Operator. Default=999

records   If successful, records retrieved.

PvsCollection[]: If successful, the PvsCollection object(s) are
                 returned.

Mcli-Get Device

Get the fields for a Device, all Devices in a Collection, Site, Farm View,
or Farm. All Devices are returned if no parameters are passed.

Get-PvsDevice

Get the fields for a Device, all Devices in a Collection, Site, Farm View,
or Farm. All Devices are returned if no parameters are passed.

-p   Parameters needed for the Get.

One of these optional

deviceId   GUID of the Device to Get.

 Guid[] Guid or DeviceId: GUID of the Device to Get.

deviceName   Name of Device to Get.

 string[] Name or DeviceName: Name of Device to Get.

deviceMac   MAC of the Device to Get.

 PvsPhysicalAddress[] DeviceMac: MAC of the Device to Get.

collectionId   GUID of the Collection to Get all Devices
                              for.

 Guid[] CollectionId: GUID of the Collection to Get all Devices for.

serverId   GUID of the Server to Get all Devices for.

 Guid[] ServerId: GUID of the Server to Get all Devices for.

serverName   Name of the Server to Get all Devices for.

 string[] ServerName: Name of the Server to Get all Devices for.

```
diskLocatorId              GUID of the DiskLocator to Get all Devices
                           for.
  Guid[] DiskLocatorId: GUID of the DiskLocator to Get all Devices for.
siteViewId                 GUID of the Site View to Get all Devices
                           for.
  Guid[] SiteViewId: GUID of the Site View to Get all Devices for.
siteId                     GUID of the Site.
  Guid[] SiteId: GUID of the Site.
siteName                   Name of the Site.
  string[] SiteName: Name of the Site.
farmViewId                 GUID of the Farm View to Get all Devices
                           for.
  Guid[] FarmViewId: GUID of the Farm View to Get all Devices for.
farmViewName               Name of the Farm View to Get all Devices
                           for.
  string[] FarmViewName: Name of the Farm View to Get all Devices for.
bdmBoot                    Include only the BDM Devices when set to 1.
                           PXE devices if set to 0. If not included,
                           all Devices are returned.
  string[] BdmBoot: Include only the BDM Devices when set to 1. PXE
                   devices if set to 0. If not included, all Devices
                   are returned.
deviceIdList               Comma delimited list of GUIDs of the Devices
                           to Get. Maximum length is 8000 characters.
deviceNameList             Comma delimited list of Names of the Devices
                           to Get. Maximum length is 8000 characters.
deviceMacList              Comma delimited list of MACs of the Devices
                           to Get. Maximum length is 8000 characters.
or one of these optional & resolutions
collectionName             Name of the Collection to Get all Devices
                           for.
  string[] CollectionName: Name of the Collection to Get all Devices
                   for.
diskLocatorName            Name of the DiskLocator to Get all Devices
                           for.
  string[] DiskLocatorName: Name of the DiskLocator to Get all Devices
                   for.
siteViewName               Name of the Site View to Get all Devices
                           for.
  string[] SiteViewName: Name of the Site View to Get all Devices for.
One of these resolutions when needed
siteId                     GUID of the Site.
  Guid[] SiteId: GUID of the Site.
```

siteName   Name of the Site.

string[] SiteName: Name of the Site.

One of these resolutions when needed

storeId   GUID of the Store that is needed when a
                                diskLocatorName is used.

*Guid[] StoreId: GUID of the Store that is needed when a
                    DiskLocatorName is used.*

storeName   Name of the Store that is needed when a
                                diskLocatorName is used.

*string[] StoreName: Name of the Store that is needed when a
                    DiskLocatorName is used.*

Instead of a parameter that matches one of the members listed

*PvsObject[] Object: PvsObjects with the members below can be used as
                the Object parameter or from a pileline:*

*DeviceName, CollectionId, ServerId, DiskLocatorId, SiteViewId,
                    SiteId or FarmViewId*

-s   Name of field to sort on. If not included,
                                the first field listed is used. Append a - to
                                the field name to get descending order.

deviceName   Computer name with no spaces. ASCII computer
                                name characters Max Length=15

description   User description. Default="" Max Length=250

deviceMac   Ethernet address as a string in the form XX-
                                XX-XX-XX-XX-XX. Uniquely identifies the
                                Device. Length=17

bootFrom   Device to boot from. Choices are 1 for
                                vDisk, 2 for Hard Disk, and 3 for Floppy.
                                This cannot be Set for a Device with
                                Personal vDisk. Min=1, Max=3, Default=1

className   Used by Automatic Update feature to match
                                new versions of Disks to a Device. This
                                cannot be Set for a Device with Personal
                                vDisk. Default="" Max Length=41

port   UDP port to use with Stream Service.
                                Min=1025, Max=65534, Default=6901

-f   List of fields to retrieve, if not included,
                                all fields are retrieved.

*If only selected fields are needed, pass them in the Fields parameter
                as a string array.*

deviceId   Read-only GUID that uniquely identifies this
                                Device. Length=36

*Guid or DeviceId: Read-only GUID that uniquely identifies this
                    Device.*

deviceName   Computer name with no spaces. ASCII computer
                                name characters Max Length=15

Name or DeviceName: Computer name with no spaces. ASCII computer name characters Max Length=15

collectionId   GUID of the Collection this Device is to be a member of. It is not used with collectionName. Length=36

CollectionId: GUID of the Collection this Device is to be a member of. It is not used with CollectionName.

collectionName   Name of the Collection this Device is to be a member of. siteName or siteId must also be used.

CollectionName: Name of the Collection this Device is to be a member of. SiteName or SiteId must also be used.

siteId   GUID of the Site the collectionName is to be a member of. This or siteName is used with collectionName.

SiteId: GUID of the Site the CollectionName is to be a member of. This or SiteName is used with CollectionName.

siteName   Name of the Site the collectionName is to be a member of. This or siteId is used with collectionName.

SiteName: Name of the Site the CollectionName is to be a member of. This or SiteId is used with CollectionName.

description   User description. Default="" Max Length=250

Description: User description. Default="" Max Length=250

deviceMac   Ethernet address as a string in the form XX-XX-XX-XX-XX-XX. Uniquely identifies the Device. Length=17

DeviceMac: Ethernet address can have the form XX-XX-XX-XX-XX-XX. Uniquely identifies the Device.

bootFrom   Device to boot from. Choices are 1 for vDisk, 2 for Hard Disk, and 3 for Floppy. This cannot be Set for a Device with Personal vDisk. Min=1, Max=3, Default=1

BootFrom: Device to boot from. Choices are 1 for vDisk, 2 for Hard Disk, and 3 for Floppy. This cannot be Set for a Device with Personal vDisk. Min=1, Max=3, Default=1

className   Used by Automatic Update feature to match new versions of Disks to a Device. This cannot be Set for a Device with Personal vDisk. Default="" Max Length=41

ClassName: Used by Automatic Update feature to match new versions of Disks to a Device. This cannot be Set for a Device with Personal vDisk. Default="" Max Length=41

port   UDP port to use with Stream Service. Min=1025, Max=65534, Default=6901

Port: UDP port to use with Stream Service. Min=1025, Max=65534, Default=6901

enabled   1 when it can be booted, 0 otherwise. This
                             cannot be Set for a Device with Personal
                             vDisk. Default=1

  Enabled: True when it can be booted, false otherwise. This cannot be
                  Set for a Device with Personal vDisk. Default=true

localDiskEnabled   If there is a local disk menu choice for the
                             Device, this is 1. This cannot be Set for a
                             Device with Personal vDisk. Default=0

  LocalDiskEnabled: If there is a local disk menu choice for the
                  Device, this is true. This cannot be Set for a
                  Device with Personal vDisk. Default=false

role   Read-only Role of the user for this item.
                             100 is Farm Administrator, 200 is Site
                             Administrator, 300 is Collection
                             Administrator, and 400 is Collection
                             Operator. Default=999

  Role: Read-only Role of the user for this item. 100 is Farm
                  Administrator, 200 is Site Administrator, 300 is
                  Collection Administrator, and 400 is Collection
                  Operator. Default=999

authentication   Device log in authentication. Choices are 0
                             for none, 1 for User Name/Password, and 2
                             for Extern. This cannot be Set for a Device
                             with Personal vDisk. Min=0, Max=2, Default=0

  Authentication: Device log in authentication. Choices are 0 for none,
                  1 for User Name/Password, and 2 for Extern. This
                  cannot be Set for a Device with Personal vDisk.
                  Min=0, Max=2, Default=0

user   Name of user to authenticate before the boot
                             process continues. This cannot be Set for a
                             Device with Personal vDisk. Default="" ASCII
                             Max Length=20

  User: Name of user to authenticate before the boot process continues.
                  This cannot be Set for a Device with Personal vDisk.
                  Default="" ASCII Max Length=20

password   Password of user to authenticate before the
                             boot process continues. This cannot be Set
                             for a Device with Personal vDisk. Default=""
                             ASCII Max Length=100

  Password: Password of user to authenticate before the boot process
                  continues. This cannot be Set for a Device with
                  Personal vDisk. Default="" ASCII Max Length=100

active   1 if the Device is currently active, 0
                             otherwise. Default=0

  Active: True if the Device is currently active, false otherwise.
                  Default=false

template   1 if the Device is the template in its
                             Collection, 0 otherwise. Default=0

adTimestamp   The time the Active Directory machine
                            account password was generated. Do not set
                            this field, it is only set internally by
                            PVS. Default=0

adSignature   The signature of the Active Directory
                            machine account password. Do not set this
                            field, it is only set internally by PVS.
                            Default=0

adPassword   The Active Directory machine account
                            password. Do not set this field, it is only
                            set internally by PVS. Default="" ASCII Max
                            Length=256

logLevel   Level to perform logging at. Values are: 0
                            (None), 1 (Fatal), 2 (Error), 3 (Warning), 4
                            (Info), 5 (Debug), and 6 (Trace). Min=0,
                            Max=6, Default=0

domainName   Fully qualified name of the domain that the
                            Device belongs to. Do not set this field, it
                            is only set internally by PVS. Default=null
                            Max Length=255

domainObjectSID   The value of the objectSID AD attribute of
                            the same name for the Device's computer
                            account. Do not set this field, it is only
                            set internally by PVS. Default=null Max
                            Length=186

domainControllerName   The name of the DC used to create the host's
                            computer account. Do not set this field, it
                            is only set internally by PVS. Default=null
                            Max Length=4000

DomainControllerName: The name of the DC used to create the host's computer account. Do not set this field, it is only set internally by PVS. Default="" Max Length=4000

domainTimeCreated   The time that the computer account was created. Do not set this field, it is only set internally by PVS. Format is: YYYY-MM-DD HH:MM:SS.SSS Default=null

DomainTimeCreated: The time that the computer account was created. Has the date and time including milliseconds. Do not set this field, it is only set internally by PVS. Default=Empty

type   1 when it performs test of Disks, 2 when it performs maintenance on Disks, 3 when it has a Personal vDisk, 4 when it has a Personal vDisk and performs tests, 0 otherwise. Min=0, Max=4, Default=0

Type: 1 when it performs test of Disks, 2 when it performs maintenance on Disks, 3 when it has a Personal vDisk, 4 when it has a Personal vDisk and performs tests, 0 otherwise. Min=0, Max=4, Default=0

pvdDriveLetter   Read-only Personal vDisk Drive letter. Range is E to U and W to Z. Default=null Max Length=1

PvdDriveLetter: Read-only Personal vDisk Drive letter. Range is E to U and W to Z. Default="" Max Length=1

localWriteCacheDiskSize   The size in GB to format the Device cache file disk. If the value is 0, then the disk is not formatted. Min=0, Max=2048, Default=0

LocalWriteCacheDiskSize: The size in GB to format the Device cache file disk. If the value is 0, then the disk is not formatted. Min=0, Max=2048, Default=0

virtualHostingPoolId   GUID that uniquely identifies the Virtual Hosting Pool for a VM. This is needed when Adding a VM device. Default=null Length=36

VirtualHostingPoolId: GUID that uniquely identifies the Virtual Hosting Pool for a VM. This is needed when Adding a VM device. Default=00000000-0000-0000-0000-000000000000

hypVmId   Hypervisor VM ID for HCL Default=null Max Length=250

HypVmId: Hypervisor VM ID for HCL Default="" Max Length=250

temporaryVersionSet   Read-only 1 when temporary version is set. Default=0

TemporaryVersionSet: Read-only true when temporary version is set. Default=false

bdmBoot   Use PXE boot when set to 0, BDM boot when set to 1. Default is PXE Default=0

            BdmBoot: Use PXE boot when set to false, BDM boot when set to true.
                         Default is PXE Default=false

     bdmType   Use PXE boot when set to 0, BDM (Bios) boot
                                 when set to 1 and BDM (Uefi) boot when set
                                 to 2. Default=0

        BdmType: Use PXE boot when set to 0, BDM (Bios) boot when set to 1
              and BDM (Uefi) boot when set to 2. Default=0

     bdmFormat   1 use VHD for BDMboot, 2 use ISO, 3 use USB.
                                 Default=0

        BdmFormat: 1 use VHD for BDMboot, 2 use ISO, 3 use USB. Default=0

     bdmUpdated   Timestamp of the last BDM boot disk update.
                                 Default=null

        BdmUpdated: Timestamp of the last BDM boot disk update. Default=Empty

     bdmCreated   Timstamp when BDM device was
                               created Default=null

        BdmCreated: Timstamp when BDM device was created Default=Empty

     xsPvsProxyUuid   UUID of XenServer PVS_proxy Default=null
                                 Length=36

        XsPvsProxyUuid: UUID of XenServer PVS_proxy Default=00000000-0000-
                   0000-0000-000000000000

   records   If successful, records retrieved.

    PvsDevice[]: If successful, the PvsDevice object(s) are returned.

Mcli-Get DeviceBootstraps

    Get all Bootstrap files for a Device, and the menuText for each.

Get-PvsDeviceBootstrap

    Get all Bootstrap files for a Device, and the MenuText for each.

   -p   Parameters needed for the Get.

    One of these required

    deviceId   GUID of the Device.

     Guid[] Guid or DeviceId: GUID of the Device.

    deviceName   Name of the Device.

     string[] Name or DeviceName: Name of the Device.

    deviceMac   MAC of the Device.

     PvsPhysicalAddress[] DeviceMac: MAC of the Device.

    Instead of a parameter that matches one of the members listed

     PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

      DeviceId

   -s   Name of field to sort on. If not included,
                           the first field listed is used. Append a - to
                           the field name to get descending order.

```
         bootstrap                    Name of the bootstrap file. Max Length=259

   -f                                 List of fields to retrieve, if not included,
                                      all fields are retrieved.

       If only selected fields are needed, pass them in the Fields parameter
                        as a string array.

       These fields exist in the DeviceBootstrap array within each
                        PvsDeviceBootstrap returned.

       Each array item is a PvsDeviceBootstrapList object.

         bootstrap                    Name of the bootstrap file. Max Length=259

          Name or Bootstrap: Name of the bootstrap file. Max Length=259

         menuText                     Text that is displayed in the Boot Menu. If
                                      this field has no value, the bootstrap value
                                      is used. Default="" ASCII Max Length=64

          MenuText: Text that is displayed in the Boot Menu. If this field has
                        no value, the bootstrap value is used. Default=""
                        ASCII Max Length=64

   records                            If successful, records retrieved.

       PvsDeviceBootstrap[]: If successful, the PvsDeviceBootstrap object(s)
                        are returned.
```

Mcli-Get DeviceDiskTempVersion

```
   Get Temporary Disk Version information for a Device, DiskLocator, Disk
      Version, Site or Farm.
```

Get-PvsDeviceDiskTempVersion

```
   Get Temporary Disk Version information for a Device, DiskLocator, Disk
      Version, Site or Farm.

   -p                                 Parameters needed for the Get.

    One of these optional

     deviceId                         GUID of the Device to get the temporary disk
                                      version information for.

        Guid[] Guid or DeviceId: GUID of the Device to get the temporary disk
                        version information for.

     deviceName                       Name of the Device to get the temporary disk
                                      version information for.

        string[] Name or DeviceName: Name of the Device to get the temporary
                        disk version information for.

     deviceMac                        MAC of the Device to get the temporary disk
                                      version information for.

        PvsPhysicalAddress[] DeviceMac: MAC of the Device to get the
                        temporary disk version information for.

     siteId                           GUID of the Site to get temporary disk
                                      version information for, and also resolution
                                      for diskLocatorName.
```

     Guid[] SiteId: GUID of the Site to get temporary disk version
             information for, and also resolution for
             DiskLocatorName.

  siteName   Name of the Site to get temporary disk
                      version information for, and also resolution
                      for diskLocatorName.

    string[] SiteName: Name of the Site to get temporary disk version
             information for, and also resolution for
             DiskLocatorName.

  diskLocatorId   GUID of the Disk Locator to get temporary
                      disk version information for.

    Guid[] DiskLocatorId: GUID of the Disk Locator to get temporary disk
             version information for.

or this optional & resolution

  diskLocatorName   Name of the Disk Locator to get temporary
                      disk version information for.

    string[] DiskLocatorName: Name of the Disk Locator to get temporary
             disk version information for.

This optional & resolution

  version   Version of the DiskLocator specified to get
                      temporary disk version information for.
                      Needs the diskLocatorId or diskLocatorName
                      too.

    string[] Version: Version of the DiskLocator specified to get
             temporary disk version information for. Needs the
             DiskLocatorId or DiskLocatorName too.

One of these resolutions when needed

  siteId   GUID of the Site.

    Guid[] SiteId: GUID of the Site to get temporary disk version
             information for, and also resolution for
             DiskLocatorName.

  siteName   Name of the Site.

    string[] SiteName: Name of the Site to get temporary disk version
             information for, and also resolution for
             DiskLocatorName.

One of these resolutions when needed

  storeId   GUID of the Store that is needed when a
                      diskLocatorName is used.

    Guid[] StoreId: GUID of the Store that is needed when a
             DiskLocatorName is used.

  storeName   Name of the Store that is needed when a
                      diskLocatorName is used.

    string[] StoreName: Name of the Store that is needed when a
             DiskLocatorName is used.

  Instead of a parameter that matches one of the members listed

PvsObject[] Object: PvsObjects with the members below can be used as the Object parameter or from a pileline:

DeviceId, SiteId or DiskLocatorId

-s   Name of field to sort on. If not included, the first field listed is used. Append a - to the field name to get descending order.

deviceName   Read-only Computer name that uniquely identifies the Device with temporary version. ASCII computer name characters

-f   List of fields to retrieve, if not included, all fields are retrieved.

If only selected fields are needed, pass them in the Fields parameter as a string array.

deviceId   Read-only GUID that uniquely identifies the Device with temporary version. Length=36

Guid or DeviceId: Read-only GUID that uniquely identifies the Device with temporary version.

deviceName   Read-only Computer name that uniquely identifies the Device with temporary version. ASCII computer name characters

Name or DeviceName: Read-only Computer name that uniquely identifies the Device with temporary version. ASCII computer name characters

diskLocatorId   Read-only GUID that uniquely identifies then Disk Locator with temporary version. Length=36

DiskLocatorId: Read-only GUID that uniquely identifies then Disk Locator with temporary version.

diskLocatorName   Read-only Name of the Disk Locator File with temporary version. It is unique within the Store. ASCII

DiskLocatorName: Read-only Name of the Disk Locator File with temporary version. It is unique within the Store. ASCII

siteId   Read-only GUID of the Site the Device and DiskLocator are a member of.

SiteId: Read-only GUID of the Site the Device and DiskLocator are a member of.

siteName   Read-only Name of the Site the Device and DiskLocator are a member of.

SiteName: Read-only Name of the Site the Device and DiskLocator are a member of.

storeId   Read-only GUID of the Store that the Disk Locator is a member of.

StoreId: Read-only GUID of the Store that the Disk Locator is a member of.

storeName   Read-only Name of the Store that the Disk
                                             Locator is a member of.

        StoreName: Read-only Name of the Store that the Disk Locator is a
                            member of.

            version   Read-only Disk version the temporary is for.

        Version: Read-only Disk version the temporary is for.

        records   If successful, records retrieved.

        PvsDeviceDiskTempVersion[]: If successful, the PvsDeviceDiskTempVersion
                        object(s) are returned.

Mcli-Get DeviceInfo

Get the fields and status for a Device, all Devices in a Collection, Site,
    Farm View, or Farm. All Devices are returned if no parameters are passed.

Get-PvsDeviceInfo

Get the fields and status for a Device, all Devices in a Collection, Site,
    Farm View, or Farm. All Devices are returned if no parameters are passed.

    -p   Parameters needed for the Get.

    One of these optional

    deviceId   GUID of the Device to Get.

        Guid[] Guid or DeviceId: GUID of the Device to Get.

    deviceName   Name of Device to Get.

        string[] Name or DeviceName: Name of Device to Get.

    deviceMac   MAC of the Device to Get.

        PvsPhysicalAddress[] DeviceMac: MAC of the Device to Get.

    collectionId   GUID of the Collection to Get all Devices
                                   for.

        Guid[] CollectionId: GUID of the Collection to Get all Devices for.

    serverId   GUID of the Server to Get all Devices for.

        Guid[] ServerId: GUID of the Server to Get all Devices for.

    serverName   Name of the Server to Get all Devices for.

        string[] ServerName: Name of the Server to Get all Devices for.

    diskLocatorId   GUID of the DiskLocator to Get all Devices
                                   for.

        Guid[] DiskLocatorId: GUID of the DiskLocator to Get all Devices for.

    siteViewId   GUID of the Site View to Get all Devices
                                   for.

        Guid[] SiteViewId: GUID of the Site View to Get all Devices for.

    siteId   GUID of the Site.

        Guid[] SiteId: GUID of the Site.

    siteName   Name of the Site.

| farmViewId | GUID of the Farm View to Get all Devices for. |

| farmViewName | Name of the Farm View to Get all Devices for. |

deviceIdList	Comma delimited list of GUIDs of the Devices to Get. Maximum length is 8000 characters.
deviceNameList	Comma delimited list of Names of the Devices to Get. Maximum length is 8000 characters.
deviceMacList	Comma delimited list of MACs of the Devices to Get. Maximum length is 8000 characters.
bdmBoot	Include only the BDM Devices when set to 1. PXE devices if set to 0. If not included, all Devices are returned.

or one of these optional & resolutions

| collectionName | Name of the Collection to Get all Devices for. |

| diskLocatorName | Name of the DiskLocator to Get all Devices for. |

| siteViewName | Name of the Site View to Get all Devices for. |

Optional

| onlyActive | Include only the active Devices when set to 1. If not included or set to 1, all Devices are returned. Only active Devices are always returned for serverId, serverName, or version. |

| makLicenseActivated | Optional MAK licensing indicator value to only return active Devices for. Values are: 0 (MAK not used), 1 (Not Activated), 2 (Activated). |

version   Version of the Disk to Get all active
                            Devices for. This is used with diskLocatorId
                            or diskLocatorName.

One of these resolutions when needed

siteId   GUID of the Site.

siteName   Name of the Site.

One of these resolutions when needed

storeId   GUID of the Store that is needed when a
                            diskLocatorName is used.

storeName   Name of the Store that is needed when a
                            diskLocatorName is used.

Instead of a parameter that matches one of the members listed

-s   Name of field to sort on. If not included,
                            the first field listed is used. Append a - to
                            the field name to get descending order.

deviceName   Computer name with no spaces. ASCII computer
                            name characters Max Length=15

description   User description. Default="" Max Length=250

deviceMac   Ethernet address as a string in the form XX-
                            XX-XX-XX-XX-XX. Uniquely identifies the
                            Device. Length=17

bootFrom   Device to boot from. Choices are 1 for
                            vDisk, 2 for Hard Disk, and 3 for Floppy.
                            This cannot be Set for a Device with
                            Personal vDisk. Min=1, Max=3, Default=1

className   Used by Automatic Update feature to match
                            new versions of Disks to a Device. This
                            cannot be Set for a Device with Personal
                            vDisk. Default="" Max Length=41

```
    port                        UDP port to use with Stream Service.
                                Min=1025, Max=65534, Default=6901

    serverName                  Read-only Name of the Server that the Device
                                is using. It is equal to "" if the Device is
                                not active.

    diskLocatorName             Read-only name of the Disk Locator File that
                                the Device is using. It is equal to the list
                                of Disk Locator names for the Device if the
                                Device is not active.

-f                              List of fields to retrieve, if not included,
                                all fields are retrieved.

    If only selected fields are needed, pass them in the Fields parameter
                        as a string array.

    deviceId                    Read-only GUID that uniquely identifies this
                                Device. Length=36

      Guid or DeviceId: Read-only GUID that uniquely identifies this
                        Device.

    deviceName                  Computer name with no spaces. ASCII computer
                                name characters Max Length=15

      Name or DeviceName: Computer name with no spaces. ASCII computer name
                        characters Max Length=15

    collectionId                GUID of the Collection this Device is to be
                                a member of. It is not used with
                                collectionName. Length=36

      CollectionId: GUID of the Collection this Device is to be a member
                        of. It is not used with CollectionName.

    collectionName              Name of the Collection this Device is to be
                                a member of. siteName or siteId must also be
                                used.

      CollectionName: Name of the Collection this Device is to be a member
                        of. SiteName or SiteId must also be used.

    siteId                      GUID of the Site the collectionName is to be
                                a member of. This or siteName is used with
                                collectionName.

      SiteId: GUID of the Site the CollectionName is to be a member of.
                        This or SiteName is used with CollectionName.

    siteName                    Name of the Site the collectionName is to be
                                a member of. This or siteId is used with
                                collectionName.

      SiteName: Name of the Site the CollectionName is to be a member of.
                        This or SiteId is used with CollectionName.

    description                 User description. Default="" Max Length=250

      Description: User description. Default="" Max Length=250

    deviceMac                   Ethernet address as a string in the form XX-
                                XX-XX-XX-XX-XX. Uniquely identifies the
                                Device. Length=17
```

DeviceMac: Ethernet address can have the form XX-XX-XX-XX-XX-XX. Uniquely identifies the Device.

bootFrom   Device to boot from. Choices are 1 for
                            vDisk, 2 for Hard Disk, and 3 for Floppy.
                            This cannot be Set for a Device with
                            Personal vDisk. Min=1, Max=3, Default=1

BootFrom: Device to boot from. Choices are 1 for vDisk, 2 for Hard Disk, and 3 for Floppy. This cannot be Set for a Device with Personal vDisk. Min=1, Max=3, Default=1

className   Used by Automatic Update feature to match
                            new versions of Disks to a Device. This
                            cannot be Set for a Device with Personal
                            vDisk. Default="" Max Length=41

ClassName: Used by Automatic Update feature to match new versions of Disks to a Device. This cannot be Set for a Device with Personal vDisk. Default="" Max Length=41

port   UDP port to use with Stream Service.
                            Min=1025, Max=65534, Default=6901

Port: UDP port to use with Stream Service. Min=1025, Max=65534, Default=6901

enabled   1 when it can be booted, 0 otherwise. This
                            cannot be Set for a Device with Personal
                            vDisk. Default=1

Enabled: True when it can be booted, false otherwise. This cannot be Set for a Device with Personal vDisk. Default=true

localDiskEnabled   If there is a local disk menu choice for the
                            Device, this is 1. This cannot be Set for a
                            Device with Personal vDisk. Default=0

LocalDiskEnabled: If there is a local disk menu choice for the Device, this is true. This cannot be Set for a Device with Personal vDisk. Default=false

role   Read-only Role of the user for this item.
                            100 is Farm Administrator, 200 is Site
                            Administrator, 300 is Collection
                            Administrator, and 400 is Collection
                            Operator. Default=999

Role: Read-only Role of the user for this item. 100 is Farm Administrator, 200 is Site Administrator, 300 is Collection Administrator, and 400 is Collection Operator. Default=999

authentication   Device log in authentication. Choices are 0
                            for none, 1 for User Name/Password, and 2
                            for Extern. This cannot be Set for a Device
                            with Personal vDisk. Min=0, Max=2, Default=0

Authentication: Device log in authentication. Choices are 0 for none, 1 for User Name/Password, and 2 for Extern. This cannot be Set for a Device with Personal vDisk. Min=0, Max=2, Default=0

| user | Name of user to authenticate before the boot process continues. This cannot be Set for a Device with Personal vDisk. Default="" ASCII Max Length=20 |

User: Name of user to authenticate before the boot process continues. This cannot be Set for a Device with Personal vDisk. Default="" ASCII Max Length=20

| password | Password of user to authenticate before the boot process continues. This cannot be Set for a Device with Personal vDisk. Default="" ASCII Max Length=100 |

Password: Password of user to authenticate before the boot process continues. This cannot be Set for a Device with Personal vDisk. Default="" ASCII Max Length=100

| active | 1 if the Device is currently active, 0 otherwise. Default=0 |

Active: True if the Device is currently active, false otherwise. Default=false

| template | 1 if the Device is the template in its Collection, 0 otherwise. Default=0 |

Template: True if the Device is the template in its Collection, false otherwise. Default=false

| adTimestamp | The time the Active Directory machine account password was generated. Do not set this field, it is only set internally by PVS. Default=0 |

AdTimestamp: The time the Active Directory machine account password was generated. Do not set this field, it is only set internally by PVS. Default=0

| adSignature | The signature of the Active Directory machine account password. Do not set this field, it is only set internally by PVS. Default=0 |

AdSignature: The signature of the Active Directory machine account password. Do not set this field, it is only set internally by PVS. Default=0

| adPassword | The Active Directory machine account password. Do not set this field, it is only set internally by PVS. Default="" ASCII Max Length=256 |

AdPassword: The Active Directory machine account password. Do not set this field, it is only set internally by PVS. Default="" ASCII Max Length=256

| logLevel | Level to perform logging at. Values are: 0 (None), 1 (Fatal), 2 (Error), 3 (Warning), 4 (Info), 5 (Debug), and 6 (Trace). Min=0, Max=6, Default=0 |

domainName   Fully qualified name of the domain that the Device belongs to. Do not set this field, it is only set internally by PVS. Default=null Max Length=255

DomainName: Fully qualified name of the domain that the Device belongs to. Do not set this field, it is only set internally by PVS. Default="" Max Length=255

domainObjectSID   The value of the objectSID AD attribute of the same name for the Device's computer account. Do not set this field, it is only set internally by PVS. Default=null Max Length=186

DomainObjectSID: The value of the objectSID AD attribute of the same name for the Device's computer account. Do not set this field, it is only set internally by PVS. Default="" Max Length=186

domainControllerName   The name of the DC used to create the host's computer account. Do not set this field, it is only set internally by PVS. Default=null Max Length=4000

DomainControllerName: The name of the DC used to create the host's computer account. Do not set this field, it is only set internally by PVS. Default="" Max Length=4000

domainTimeCreated   The time that the computer account was created. Do not set this field, it is only set internally by PVS. Format is: YYYY-MM-DD HH:MM:SS.SSS Default=null

DomainTimeCreated: The time that the computer account was created. Has the date and time including milliseconds. Do not set this field, it is only set internally by PVS. Default=Empty

type   1 when it performs test of Disks, 2 when it performs maintenance on Disks, 3 when it has a Personal vDisk, 4 when it has a Personal vDisk and performs tests, 0 otherwise. Min=0, Max=4, Default=0

Type: 1 when it performs test of Disks, 2 when it performs maintenance on Disks, 3 when it has a Personal vDisk, 4 when it has a Personal vDisk and performs tests, 0 otherwise. Min=0, Max=4, Default=0

pvdDriveLetter   Read-only Personal vDisk Drive letter. Range is E to U and W to Z. Default=null Max Length=1

PvdDriveLetter: Read-only Personal vDisk Drive letter. Range is E to U and W to Z. Default="" Max Length=1

localWriteCacheDiskSize   The size in GB to format the Device cache
                            file disk. If the value is 0, then the disk
                            is not formatted. Min=0, Max=2048, Default=0

LocalWriteCacheDiskSize: The size in GB to format the Device cache
                file disk. If the value is 0, then the disk is not
                formatted. Min=0, Max=2048, Default=0

virtualHostingPoolId   GUID that uniquely identifies the Virtual
                            Hosting Pool for a VM. This is needed when
                            Adding a VM device. Default=null Length=36

VirtualHostingPoolId: GUID that uniquely identifies the Virtual
                Hosting Pool for a VM. This is needed when Adding a
                VM device. Default=00000000-0000-0000-0000-
                000000000000

hypVmId   Hypervisor VM ID for HCL Default=null Max
                            Length=250

HypVmId: Hypervisor VM ID for HCL Default="" Max Length=250

temporaryVersionSet   Read-only 1 when temporary version is set.
                            Default=0

TemporaryVersionSet: Read-only true when temporary version is set.
                Default=false

bdmBoot   Use PXE boot when set to 0, BDM boot when
                            set to 1. Default is PXE Default=0

BdmBoot: Use PXE boot when set to false, BDM boot when set to true.
                Default is PXE Default=false

bdmType   Use PXE boot when set to 0, BDM (Bios) boot
                            when set to 1 and BDM (Uefi) boot when set
                            to 2. Default=0

BdmType: Use PXE boot when set to 0, BDM (Bios) boot when set to 1
                and BDM (Uefi) boot when set to 2. Default=0

bdmFormat   1 use VHD for BDMboot, 2 use ISO, 3 use USB.
                            Default=0

BdmFormat: 1 use VHD for BDMboot, 2 use ISO, 3 use USB. Default=0

bdmUpdated   Timestamp of the last BDM boot disk update.
                            Default=null

BdmUpdated: Timestamp of the last BDM boot disk update. Default=Empty

bdmCreated   Timstamp when BDM device was
                            created Default=null

BdmCreated: Timstamp when BDM device was created Default=Empty

xsPvsProxyUuid   UUID of XenServer PVS_proxy Default=null
                            Length=36

XsPvsProxyUuid: UUID of XenServer PVS_proxy Default=00000000-0000-
                0000-0000-000000000000

ip   Read-only IP of the Device. It is equal to
                            "" if the Device is not active.

Ip: Read-only IP of the Device. It is equal to 0.0.0.0 if the Device
                is not active.

serverPortConnection   Read-only Port of the Server that the Device is using. It is equal to "" if the Device is not active. Default=0

ServerPortConnection: Read-only Port of the Server that the Device is using. It is equal to 0 if the Device is not active. Default=0

serverIpConnection   Read-only IP of the Server that the Device is using. It is equal to "" if the Device is not active.

ServerIpConnection: Read-only IP of the Server that the Device is using. It is equal to 0.0.0.0 if the Device is not active.

serverId   Read-only GUID of the Server that the Device is using. It is equal to "" if the Device is not active. Length=36

ServerId: Read-only GUID of the Server that the Device is using. It is equal to 00000000-0000-0000-0000-000000000000 if the Device is not active.

serverName   Read-only Name of the Server that the Device is using. It is equal to "" if the Device is not active.

ServerName: Read-only Name of the Server that the Device is using. It is equal to "" if the Device is not active.

diskLocatorId   Read-only GUID of the Disk Locator that the Device is using. It is equal to "" if the Device is not active. Length=36

DiskLocatorId: Read-only GUID of the Disk Locator that the Device is using. It is equal to 00000000-0000-0000-0000-000000000000 if the Device is not active.

diskLocatorName   Read-only name of the Disk Locator File that the Device is using. It is equal to the list of Disk Locator names for the Device if the Device is not active.

DiskLocatorName: Read-only name of the Disk Locator File that the Device is using. It is equal to the list of Disk Locator names for the Device if the Device is not active.

diskVersion   Read-only version of the Disk Locator File that the Device is using. It is equal to "" if the Device is not active. Default=0

DiskVersion: Read-only version of the Disk Locator File that the Device is using. It is equal to 0 if the Device is not active. Default=0

diskVersionAccess   State of the Disk Version. Values are: 0 (Production), 1 (Maintenance), 2 (MaintenanceHighestVersion), 3 (Override), 4 (Merge), 5 (MergeMaintenance), 6 (MergeTest), and 7 (Test). It is equal to "" if the Device is not active. Default=0

DiskVersionAccess: State of the Disk Version. Values are: 0
                   (Production), 1 (Maintenance), 2
                   (MaintenanceHighestVersion), 3 (Override), 4
                   (Merge), 5 (MergeMaintenance), 6 (MergeTest), and 7
                   (Test). It is equal to 0 if the Device is not
                   active. Default=0

diskFileName   Name of the Disk File including the
                        extension. It is equal to "" if the Device
                        is not active.

DiskFileName: Name of the Disk File including the extension. It is
              equal to "" if the Device is not active.

status   1 or 2 numbers in the format n,n. They are
                        the number of retries and if ram cache is
                        being used, ram cache percent used. It is
                        equal to "" if the Device is not active.

Status: 1 or 2 numbers in the format n,n. They are the number of
        retries and if ram cache is being used, ram cache
        percent used. It is equal to "" if the Device is not
        active.

licenseType   0 when None, 1 for Desktop, 2 for Server, 5
                        for OEM SmartClient, 6 for XenApp, 7 for
                        XenDesktop. It is equal to 0 if the Device
                        is not active. Default=0

LicenseType: 0 when None, 1 for Desktop, 2 for Server, 5 for OEM
             SmartClient, 6 for XenApp, 7 for XenDesktop. It is
             equal to 0 if the Device is not active. Default=0

makLicenseActivated   Read-only indicator if MAK licensing is
                        being used and is activated. Values are: 0
                        (MAK not used), 1 (Not Activated), 2
                        (Activated). It is equal to "" if the Device
                        is not active. Default=0

MakLicenseActivated: Read-only indicator if MAK licensing is being
                     used and is activated. Values are: 0 (MAK not used),
                     1 (Not Activated), 2 (Activated). It is equal to 0
                     if the Device is not active. Default=0

model   Oem Only: Read-only model of the computer.
                        Values are OptiPlex 745, 755, 320, 760,
                        FX160, or Default. It is equal to "" if the
                        Device is not active.

Model: Oem Only: Read-only model of the computer. Values are OptiPlex
       745, 755, 320, 760, FX160, or Default. It is equal
       to "" if the Device is not active.

license   Oem Only: Read-only type of the license.
                        Values are 0 when None, 1 or 2 when Desktop.
                        It is equal to 0 if the Device is not
                        active. Default=0

License: Oem Only: Read-only type of the license. Values are 0 when
         None, 1 or 2 when Desktop. It is equal to 0 if the
         Device is not active. Default=0

records   If successful, records retrieved.

PvsDeviceInfo[]: If successful, the PvsDeviceInfo object(s) are
returned.

Mcli-Get DevicePersonality

Get the Device Personality names and values.

Get-PvsDevicePersonality

Get the Device Personality names and values.

  -p   Parameters needed for the Get.

  One of these required

   deviceId   GUID of the Device.

    Guid[] Guid or DeviceId: GUID of the Device.

   deviceName   Name of the Device.

    string[] Name or DeviceName: Name of the Device.

   deviceMac   MAC of the Device.

    PvsPhysicalAddress[] DeviceMac: MAC of the Device.

   Instead of a parameter that matches one of the members listed

    PvsObject[] Object: PvsObjects with the members below can be used as
the Object parameter or from a pileline:

     DeviceId

  -s   Name of field to sort on. If not included,
the first field listed is used. Append a - to
the field name to get descending order.

   name   Name of the Device personality item. Max
Length=250

  -f   List of fields to retrieve, if not included,
all fields are retrieved.

  If only selected fields are needed, pass them in the Fields parameter
as a string array.

  These fields exist in the DevicePersonality array within each
PvsDevicePersonality returned.

  Each array item is a PvsDevicePersonalityList object.

   name   Name of the Device personality item. Max
Length=250

    Name: Name of the Device personality item. Max Length=250

   value   Value for the Device personality item. Max
Length=1000

    Value: Value for the Device personality item. Max Length=1000

records   If successful, records retrieved.

  PvsDevicePersonality[]: If successful, the PvsDevicePersonality
object(s) are returned.

Mcli-Get DeviceStatus

Get the DeviceStatus fields for a Device or all Devices for a Server, Disk Locator, or Farm. All Devices are returned if no parameters are passed.

Get-PvsDeviceStatus

Get the DeviceStatus fields for a Device or all Devices for a Server, Disk Locator, or Farm. All Devices are returned if no parameters are passed.

-p   Parameters needed for the Get.

 One of these optional

  deviceId   GUID of the Device to Get status for.

    Guid[] Guid or DeviceId: GUID of the Device to Get status for.

  deviceName   Name of Device to Get status for.

    string[] Name or DeviceName: Name of Device to Get status for.

  deviceMac   MAC of the Device to Get status for.

    PvsPhysicalAddress[] DeviceMac: MAC of the Device to Get status for.

  serverId   GUID of the Server to Get all Device Status
                                for.

    Guid[] ServerId: GUID of the Server to Get all Device Status for.

  serverName   Name of the Server to Get all Device Status
                                for.

    string[] ServerName: Name of the Server to Get all Device Status for.

  diskLocatorId   GUID of the Disk Locator to Get all
                                DeviceStatus for.

    Guid[] DiskLocatorId: GUID of the Disk Locator to Get all
                    DeviceStatus for.

  collectionId   GUID of the Collection to Get all
                                DeviceStatus for.

    Guid[] CollectionId: GUID of the Collection to Get all DeviceStatus
                    for.

 or one of these optional & resolutions

  diskLocatorName   Name of the Disk Locator File to Get all
                                DeviceStatus for.

    string[] DiskLocatorName: Name of the Disk Locator File to Get all
                    DeviceStatus for.

  collectionName   Name of the Collection to Get all
                                DeviceStatus for.

    string[] CollectionName: Name of the Collection to Get all
                    DeviceStatus for.

 One of these resolutions when needed

  siteId   GUID of the Site.

    Guid[] SiteId: GUID of the Site.

  siteName   Name of the Site.

string[] SiteName: Name of the Site.

One of these resolutions when needed

storeId   GUID of the Store that is needed when a diskLocatorName is used.

Guid[] StoreId: GUID of the Store that is needed when a DiskLocatorName is used.

storeName   Name of the Store that is needed when a diskLocatorName is used.

string[] StoreName: Name of the Store that is needed when a DiskLocatorName is used.

Instead of a parameter that matches one of the members listed

PvsObject[] Object: PvsObjects with the members below can be used as the Object parameter or from a pileline:

DeviceId, ServerId, DiskLocatorId or CollectionId

-s   Name of field to sort on. If not included, the first field listed is used. Append a - to the field name to get descending order.

deviceName   Read-only Name of the Device. Can be used with Get Device.

-f   List of fields to retrieve, if not included, all fields are retrieved.

If only selected fields are needed, pass them in the Fields parameter as a string array.

deviceId   Read-only GUID of the Device. Can be used with Get Device. Length=36

Guid or DeviceId: Read-only GUID of the Device. Can be used with Get Device.

deviceName   Read-only Name of the Device. Can be used with Get Device.

Name or DeviceName: Read-only Name of the Device. Can be used with Get Device.

ip   Read-only IP of the Device.

Ip: Read-only IP of the Device.

serverPortConnection   Read-only Port of the Server that the Device is using. Default=0

ServerPortConnection: Read-only Port of the Server that the Device is using. Default=0

serverIpConnection   Read-only IP of the Server that the Device is using.

ServerIpConnection: Read-only IP of the Server that the Device is using.

serverId   Read-only GUID of the Server that the Device is using. Length=36

ServerId: Read-only GUID of the Server that the Device is using.

| serverName | Read-only Name of the Server that the Device is using. |

ServerName: Read-only Name of the Server that the Device is using.

| diskLocatorId | Read-only GUID of the Disk Locator that the Device is using. Length=36 |

DiskLocatorId: Read-only GUID of the Disk Locator that the Device is using.

| diskLocatorName | Read-only name of the Disk Locator File that the Device is using. |

DiskLocatorName: Read-only name of the Disk Locator File that the Device is using.

| diskVersion | Read-only version of the Disk Locator File that the Device is using. Default=0 |

DiskVersion: Read-only version of the Disk Locator File that the Device is using. Default=0

| diskVersionAccess | State of the Disk Version. Values are: 0 (Production), 1 (Maintenance), 2 (MaintenanceHighestVersion), 3 (Override), 4 (Merge), 5 (MergeMaintenance), 6 (MergeTest), and 7 (Test) Default=0 |

DiskVersionAccess: State of the Disk Version. Values are: 0 (Production), 1 (Maintenance), 2 (MaintenanceHighestVersion), 3 (Override), 4 (Merge), 5 (MergeMaintenance), 6 (MergeTest), and 7 (Test) Default=0

| diskFileName | Name of the Disk File including the extension. |

DiskFileName: Name of the Disk File including the extension.

| status | 1 or 2 numbers in the format n,n. They are the number of retries and if ram cache is being used, ram cache percent used. |

Status: 1 or 2 numbers in the format n,n. They are the number of retries and if ram cache is being used, ram cache percent used.

| licenseType | 0 when None, 1 for Desktop, 2 for Server, 5 for OEM SmartClient, 6 for XenApp, 7 for XenDesktop. Default=0 |

LicenseType: 0 when None, 1 for Desktop, 2 for Server, 5 for OEM SmartClient, 6 for XenApp, 7 for XenDesktop. Default=0

| makLicenseActivated | Read-only indicator if MAK licensing is being used and is activated. Values are: 0 (MAK not used), 1 (Not Activated), 2 (Activated). Default=0 |

MakLicenseActivated: Read-only indicator if MAK licensing is being used and is activated. Values are: 0 (MAK not used), 1 (Not Activated), 2 (Activated). Default=0

```
      records                            If successful, records retrieved.

          PvsDeviceStatus[]: If successful, the PvsDeviceStatus object(s) are
                             returned.
```

Mcli-Get Disk

```
   Get the fields for a single disk.
```

Get-PvsDisk

```
   Get the fields for a single disk.

     -p                                Parameters needed for the Get.
       This required
        diskLocatorId           GUID of the Disk Locator.

          Guid[] Guid or DiskLocatorId: GUID of the Disk Locator.
        or this required & resolution
        diskLocatorName               Name of the Disk Locator.

          string[] Name or DiskLocatorName: Name of the Disk Locator.
       One of these resolutions when needed
        siteId                        GUID of the Site.

          Guid[] SiteId: GUID of the Site.
        siteName                      Name of the Site.

          string[] SiteName: Name of the Site.
       One of these resolutions when needed
        storeId                        GUID of the Store that is needed when a
                                       diskLocatorName is used.

          Guid[] StoreId: GUID of the Store that is needed when a
                          DiskLocatorName is used.
        storeName                      Name of the Store that is needed when a
                                       diskLocatorName is used.

          string[] StoreName: Name of the Store that is needed when a
                              DiskLocatorName is used.
       Instead of a parameter that matches one of the members listed

          PvsObject[] Object: PvsObjects with the members below can be used as
                              the Object parameter or from a pileline:

            DiskLocatorId
     -f                                List of fields to retrieve, if not included,
                                       all fields are retrieved.

       If only selected fields are needed, pass them in the Fields parameter
                          as a string array.
        class                          Class of the Disk. Max Length=40

          Class: Class of the Disk. Max Length=40

        imageType                      Type of this image (software type). Max
                                       Length=40
```

ImageType: Type of this image (software type). Max Length=40

diskSize   Read-only size of the image. The value is 0 when it is not available. Default=0

DiskSize: Read-only size of the image. The value is 0 when it is not available. Default=0

vhdBlockSize   Block size in KB. For VHD it is only used with Dynamic type. Tested sizes for VHD are 512, 2048, and 16384. VHD Min=512, Max=16384, Default=2048. For VHDX it is used for all types. Tested size for VHDX is 32768. VHDX Min=1024, Max= 262144, Default=32768. Default=0

VhdBlockSize: Block size in KB. For VHD it is only used with Dynamic type. Tested sizes for VHD are 512, 2048, and 16384. VHD Min=512, Max=16384, Default=2048. For VHDX it is used for all types. Tested size for VHDX is 32768. VHDX Min=1024, Max= 262144, Default=32768. Default=0

logicalSectorSize   Logical Sector Size. Values are: 512, 4096, Default=512

LogicalSectorSize: Logical Sector Size. Values are: 512, 4096, Default=512

writeCacheSize   RAM cache size (MB). Not 0 when used with Cache in Device RAM, and Cache in Device RAM with Overflow on Hard Disk. A value of 0 will disable the RAM use for Cache in Device RAM with Overflow on Hard Disk. Min=0, Max=131072, Default=0

WriteCacheSize: RAM cache size (MB). Not 0 when used with Cache in Device RAM, and Cache in Device RAM with Overflow on Hard Disk. A value of 0 will disable the RAM use for Cache in Device RAM with Overflow on Hard Disk. Min=0, Max=131072, Default=0

autoUpdateEnabled   Automatically update this image for matching Devices when set to 1. Default 0

AutoUpdateEnabled: Automatically update this image for matching Devices when set to true. Default false

activationDateEnabled   Use activation date to activate image when set to 1. Default 0

ActivationDateEnabled: Use activation date to activate image when set to true. Default false

adPasswordEnabled   Enable AD password management when set to 1.

AdPasswordEnabled: Enable AD password management when set to true.

haEnabled   Enable HA when set to 1.

HaEnabled: Enable HA when set to true.

printerManagementEnabled   Invalid printers will be deleted from the Device when set to 1.

writeCacheType	0 (Private), (other values are standard image) 1 (Cache on Server), 3 (Cache in Device RAM), 4 (Cache on Device Hard Disk), 7 (Cache on Server, Persistent), or 9 (Cache in Device RAM with Overflow on Hard Disk). Min=0, Max=9, Default=0

WriteCacheType: 0 (Private), (other values are standard image) 1 (Cache on Server), 3 (Cache in Device RAM), 4 (Cache on Device Hard Disk), 7 (Cache on Server, Persistent), or 9 (Cache in Device RAM with Overflow on Hard Disk). Min=0, Max=9, Default=0

licenseMode	0 (None), 1 (Multiple Activation Key), or 2 (Key Management Service). Min=0, Max=2, Default=0

LicenseMode: 0 (None), 1 (Multiple Activation Key), or 2 (Key Management Service). Min=0, Max=2, Default=0

activeDate	Date to activate the disk if autoUpdateEnabled and activationDateEnabled are 1. Format is yyyy/mm/dd. Empty when the autoUpdateEnabled or activationDateEnabled are 0.

ActiveDate: Date to activate the disk if AutoUpdateEnabled and activationDateEnabled are true. Has the date. Empty when the AutoUpdateEnabled or activationDateEnabled are false.

longDescription	Description of the Disk. Max Length=399

LongDescription: Description of the Disk. Max Length=399

operatingSystem	Operating System of Disk. Max Length=250

OperatingSystem: Operating System of Disk. Max Length=250

osType	Operating System Type of Disk. Max Length=40

OsType: Operating System Type of Disk. Max Length=40

serialNumber	User defined serial number. Max Length=36

SerialNumber: User defined serial number. Max Length=36

date	User defined date. Max Length=40

Date: User defined date. Max Length=40

author	User defined author. Max Length=40

Author: User defined author. Max Length=40

title	User defined title. Max Length=40

Title: User defined title. Max Length=40

company	User defined company. Max Length=40

Company: User defined company. Max Length=40

internalName	User defined name. Max Length=63

       InternalName: User defined name. Max Length=63

originalFile   User defined original file. Max Length=127

   OriginalFile: User defined original file. Max Length=127

hardwareTarget   User defined hardware target. Max Length=127

   HardwareTarget: User defined hardware target. Max Length=127

majorRelease   User defined major release number. Min=0, Max=4294967295, Default=0

   MajorRelease: User defined major release number. Min=0, Max=4294967295, Default=0

minorRelease   User defined minor release number. Min=0, Max=4294967295, Default=0

   MinorRelease: User defined minor release number. Min=0, Max=4294967295, Default=0

build   User defined build number. Min=0, Max=4294967295, Default=0

   Build: User defined build number. Min=0, Max=4294967295, Default=0

clearCacheDisabled   Clear cached secrets disabled.

   ClearCacheDisabled: Clear cached secrets disabled.

format   Format of the image. Values are: 0 (VHD), 1 (VHDX). Default=0

   VHDX: If VHDX is true, the format of the image is VHDX. Otherwise it is VHD. Default=false

records   If successful, records retrieved.

   PvsDisk[]: If successful, the PvsDisk object(s) are returned.

Mcli-Get DiskInfo

Get the fields for a Disk and Disk Locator or all Disks and Disk Locators for a Device, Server, Store, Site, or Farm. All Disks and DiskLocators are returned if no parameters are passed.

Get-PvsDiskInfo

Get the fields for a Disk and Disk Locator or all Disks and Disk Locators for a Device, Server, Store, Site, or Farm. All Disks and DiskLocators are returned if no parameters are passed.

-p   Parameters needed for the Get.

 One of these optional

diskLocatorId   GUID of the Disk Locator to Get.

   Guid[] Guid or DiskLocatorId: GUID of the Disk Locator to Get.

deviceId   GUID of the Device to Get all DiskLocators for.

   Guid[] DeviceId: GUID of the Device to Get all DiskLocators for.

deviceName   Name of the Device to Get all DiskLocators for.

string[] DeviceName: Name of the Device to Get all DiskLocators for.

deviceMac   MAC of the Device to Get all DiskLocators for.

PvsPhysicalAddress[] DeviceMac: MAC of the Device to Get all DiskLocators for.

serverId   GUID of the Server to Get all DiskLocators for.

Guid[] ServerId: GUID of the Server to Get all DiskLocators for.

serverName   Name of the Server to Get all DiskLocators for.

string[] ServerName: Name of the Server to Get all DiskLocators for.

updateTaskId   GUID of the Update Task to Get all DiskLocators for.

Guid[] UpdateTaskId: GUID of the Update Task to Get all DiskLocators for.

siteId   GUID of the Site to Get all DiskLocators for.

Guid[] SiteId: GUID of the Site to Get all DiskLocators for.

siteName   Name of the Site to Get all DiskLocators for.

string[] SiteName: Name of the Site to Get all DiskLocators for.

or one of these optional & resolutions

diskLocatorName   Name of the Disk Locator File to Get.

string[] Name or DiskLocatorName: Name of the Disk Locator File to Get.

storeId   GUID of the Store to Get all DiskLocators for.

Guid[] StoreId: GUID of the Store to Get all DiskLocators for.

storeName   Name of the Store to Get all DiskLocators for.

string[] StoreName: Name of the Store to Get all DiskLocators for.

updateTaskName   Name of the Update Task to Get all DiskLocators for.

string[] UpdateTaskName: Name of the Update Task to Get all DiskLocators for.

One of these optional

single   Include single server connection when set to 1. If this and All are not included, both connection types are included.

SwitchParameter Single: If -Single is specified, include single server connection. If this and All are not included, both connection types are included.

```
       all                       Include all server connections for the store
                                  when set to 1. If this and Single are not
                                  included, both connection types are
                                  included.
```

SwitchParameter All: If -All is specified, include all server
                     connections for the store. If this and Single are
                     not included, both connection types are included.

```
    Optional
```

```
     onlyActive                  Include only the active DiskLocators when
                                  set to 1. If not included or set to 0, all
                                  DiskLocators are returned.
```

SwitchParameter OnlyActive: If -OnlyActive is specified, include only
                     the active DiskLocators. If not included, all
                     DiskLocators are returned.

```
     updateDevice                Include DiskLocators that have an Update
                                  Device when set to 1. Include DiskLocators
                                  that do not have an Update Device when set
                                  to 0. If this parameter is not included,
                                  then all DiskLocators are returned.
```

SwitchParameter UpdateDevice: If -UpdateDevice is specified, include
                     only DiskLocators that have an Update Device.

```
    One of these resolutions when needed
```

```
     siteId                      GUID of the Site.
```

Guid[] SiteId: GUID of the Site to Get all DiskLocators for.

```
     siteName                    Name of the Site.
```

string[] SiteName: Name of the Site to Get all DiskLocators for.

```
    One of these resolutions when needed
```

```
     storeId                     GUID of the Store that is needed when a
                                  diskLocatorName is used.
```

Guid[] StoreId: GUID of the Store to Get all DiskLocators for.

```
     storeName                   Name of the Store that is needed when a
                                  diskLocatorName is used.
```

string[] StoreName: Name of the Store to Get all DiskLocators for.

```
    Instead of a parameter that matches one of the members listed
```

PvsObject[] Object: PvsObjects with the members below can be used as
                     the Object parameter or from a pileline:

DiskLocatorId, DeviceId, ServerId, UpdateTaskId or SiteId

```
 -s                          Name of field to sort on. If not included,
                             the first field listed is used. Append a - to
                             the field name to get descending order.
```

```
     diskLocatorName             Name of the Disk Locator File. It is unique
                                  within the Store. ASCII Max Length=52
```

```
     description                 User description. Default="" Max Length=250
```

menuText   Text that is displayed in the Boot Menu. If
                                  this field has no value, the name value is
                                  used. Default="" ASCII Max Length=64

-f   List of fields to retrieve, if not included,
                                  all fields are retrieved.

*If only selected fields are needed, pass them in the Fields parameter
                        as a string array.*

  diskLocatorId   Read-only GUID that uniquely identifies this
                                  Disk Locator. Length=36

   *Guid or DiskLocatorId: Read-only GUID that uniquely identifies this
                        Disk Locator.*

  diskLocatorName   Name of the Disk Locator File. It is unique
                                  within the Store. ASCII Max Length=52

    *Name or DiskLocatorName: Name of the Disk Locator File. It is unique
                     within the Store. ASCII Max Length=52*

  siteId   GUID of the Site this DiskLocator is to be a
                                  member of. It is not used with siteName.
                                  Length=36

     *SiteId: GUID of the Site this DiskLocator is to be a member of. It is
                     not used with SiteName.*

  siteName   Name of the Site this DiskLocator is to be a
                                  member of. It is not used with siteId.

    *SiteName: Name of the Site this DiskLocator is to be a member of. It
                     is not used with SiteId.*

  storeId   GUID of the Store that this Disk Locator is
                                  a member of. siteName or siteId must also be
                                  used. It is not used with storeName.
                                  Length=36

    *StoreId: GUID of the Store that this Disk Locator is a member of.
                     SiteName or SiteId must also be used. It is not used
                     with StoreName.*

  storeName   Name of the Store that this Disk Locator is
                                  a member of. siteName or siteId must also be
                                  used. It is not used with storeId.

    *StoreName: Name of the Store that this Disk Locator is a member of.
                     SiteName or SiteId must also be used. It is not used
                     with StoreId.*

  description   User description. Default="" Max Length=250

    Description: User description. Default="" Max Length=250

  menuText   Text that is displayed in the Boot Menu. If
                                  this field has no value, the name value is
                                  used. Default="" ASCII Max Length=64

    *MenuText: Text that is displayed in the Boot Menu. If this field has
                     no value, the name value is used. Default="" ASCII
                     Max Length=64*

serverId	GUID of the single Server that this Disk Locator is assigned to. It is not used with serverName. Default=null Length=36

    ServerId: GUID of the single Server that this Disk Locator is assigned to. It is not used with ServerName. Default=00000000-0000-0000-0000-000000000000

serverName	Name of the single Server that this Disk Locator is assigned to. It is not used with serverId. Default=null

    ServerName: Name of the single Server that this Disk Locator is assigned to. It is not used with ServerId. Default=""

enabled	1 when this disk can be booted, 0 otherwise. Default=1

    Enabled: True when this disk can be booted, false otherwise. Default=true

role	Read-only Role of the user for this item. 100 is Farm Administrator, 200 is Site Administrator, 300 is Collection Administrator, and 999 is read-only. Default=999

    Role: Read-only Role of the user for this item. 100 is Farm Administrator, 200 is Site Administrator, 300 is Collection Administrator, and 999 is read-only. Default=999

mapped	1 if the Disk is currently mapped, 0 otherwise. Default=0

    Mapped: True if the Disk is currently mapped, false otherwise. Default=false

enabledForDevice	1 when this disk is enabled for the Device specified, 0 otherwise. This is only returned when a Device is specified. Default=1

    EnabledForDevice: True when this disk is enabled for the Device specified, false otherwise. This is only returned when a Device is specified. Default=true

active	1 if the DiskLocator is currently active, 0 otherwise. Default=0

    Active: True if the DiskLocator is currently active, false otherwise. Default=false

rebalanceEnabled	1 when this Server can automatically rebalance Devices, 0 otherwise. Default=0

    RebalanceEnabled: True when this Server can automatically rebalance Devices, false otherwise. Default=false

rebalanceTriggerPercent	Percent over fair load that triggers a dynamic Device rebalance. Min=5, Max=5000, Default=25

RebalanceTriggerPercent: Percent over fair load that triggers a dynamic Device rebalance. Min=5, Max=5000, Default=25

subnetAffinity   Qualifier for subnet affinity when assigning a Server. 0=None, 1=Best Effort, 2=Fixed. Min=0, Max=2, Default=0

SubnetAffinity: Qualifier for subnet affinity when assigning a Server. 0=None, 1=Best Effort, 2=Fixed. Min=0, Max=2, Default=0

diskUpdateDeviceId   GUID of the DiskUpdateDevice that is used when updates are performed. Default=null

DiskUpdateDeviceId: GUID of the DiskUpdateDevice that is used when updates are performed. Default=00000000-0000-0000-0000-000000000000

diskUpdateDeviceName   Name of the DiskUpdateDevice that is used when updates are performed. Default=null

DiskUpdateDeviceName: Name of the DiskUpdateDevice that is used when updates are performed. Default=""

temporaryVersionSet   Read-only 1 when temporary version(s) are set. Default=0

TemporaryVersionSet: Read-only true when temporary version(s) are set. Default=false

class   Class of the Disk. Max Length=40

Class: Class of the Disk. Max Length=40

imageType   Type of this image (software type). Max Length=40

ImageType: Type of this image (software type). Max Length=40

diskSize   Read-only size of the image. The value is 0 when it is not available. Default=0

DiskSize: Read-only size of the image. The value is 0 when it is not available. Default=0

vhdBlockSize   Block size in KB. For VHD it is only used with Dynamic type. Tested sizes for VHD are 512, 2048, and 16384. VHD Min=512, Max=16384, Default=2048. For VHDX it is used for all types. Tested size for VHDX is 32768. VHDX Min=1024, Max= 262144, Default=32768. Default=0

VhdBlockSize: Block size in KB. For VHD it is only used with Dynamic type. Tested sizes for VHD are 512, 2048, and 16384. VHD Min=512, Max=16384, Default=2048. For VHDX it is used for all types. Tested size for VHDX is 32768. VHDX Min=1024, Max= 262144, Default=32768. Default=0

logicalSectorSize   Logical Sector Size. Values are: 512, 4096, Default=512

LogicalSectorSize: Logical Sector Size. Values are: 512, 4096, Default=512

writeCacheSize	RAM cache size (MB). Not 0 when used with Cache in Device RAM, and Cache in Device RAM with Overflow on Hard Disk. A value of 0 will disable the RAM use for Cache in Device RAM with Overflow on Hard Disk. Min=0, Max=131072, Default=0

WriteCacheSize: RAM cache size (MB). Not 0 when used with Cache in Device RAM, and Cache in Device RAM with Overflow on Hard Disk. A value of 0 will disable the RAM use for Cache in Device RAM with Overflow on Hard Disk. Min=0, Max=131072, Default=0

autoUpdateEnabled	Automatically update this image for matching Devices when set to 1. Default 0

AutoUpdateEnabled: Automatically update this image for matching Devices when set to true. Default false

activationDateEnabled	Use activation date to activate image when set to 1. Default 0

ActivationDateEnabled: Use activation date to activate image when set to true. Default false

adPasswordEnabled	Enable AD password management when set to 1.

AdPasswordEnabled: Enable AD password management when set to true.

haEnabled	Enable HA when set to 1.

HaEnabled: Enable HA when set to true.

printerManagementEnabled	Invalid printers will be deleted from the Device when set to 1.

PrinterManagementEnabled: Invalid printers will be deleted from the Device when set to true.

writeCacheType	0 (Private), (other values are standard image) 1 (Cache on Server), 3 (Cache in Device RAM), 4 (Cache on Device Hard Disk), 7 (Cache on Server, Persistent), or 9 (Cache in Device RAM with Overflow on Hard Disk). Min=0, Max=9, Default=0

WriteCacheType: 0 (Private), (other values are standard image) 1 (Cache on Server), 3 (Cache in Device RAM), 4 (Cache on Device Hard Disk), 7 (Cache on Server, Persistent), or 9 (Cache in Device RAM with Overflow on Hard Disk). Min=0, Max=9, Default=0

licenseMode	0 (None), 1 (Multiple Activation Key), or 2 (Key Management Service). Min=0, Max=2, Default=0

LicenseMode: 0 (None), 1 (Multiple Activation Key), or 2 (Key Management Service). Min=0, Max=2, Default=0

activeDate	Date to activate the disk if autoUpdateEnabled and activationDateEnabled are 1. Format is yyyy/mm/dd. Empty when the autoUpdateEnabled or activationDateEnabled are 0.

ActiveDate: Date to activate the disk if AutoUpdateEnabled and
                activationDateEnabled are true. Has the date. Empty
                when the AutoUpdateEnabled or activationDateEnabled
                are false.
longDescription   Description of the Disk. Max Length=399
 LongDescription: Description of the Disk. Max Length=399
operatingSystem   Operating System of Disk. Max Length=250
 OperatingSystem: Operating System of Disk. Max Length=250
osType   Operating System Type of Disk. Max Length=40
 OsType: Operating System Type of Disk. Max Length=40
serialNumber   User defined serial number. Max Length=36
 SerialNumber: User defined serial number. Max Length=36
date   User defined date. Max Length=40
 Date: User defined date. Max Length=40
author   User defined author. Max Length=40
 Author: User defined author. Max Length=40
title   User defined title. Max Length=40
 Title: User defined title. Max Length=40
company   User defined company. Max Length=40
 Company: User defined company. Max Length=40
internalName   User defined name. Max Length=63
 InternalName: User defined name. Max Length=63
originalFile   User defined original file. Max Length=127
 OriginalFile: User defined original file. Max Length=127
hardwareTarget   User defined hardware target. Max Length=127
 HardwareTarget: User defined hardware target. Max Length=127
majorRelease   User defined major release number. Min=0,
                         Max=4294967295, Default=0
 MajorRelease: User defined major release number. Min=0,
               Max=4294967295, Default=0
minorRelease   User defined minor release number. Min=0,
                         Max=4294967295, Default=0
 MinorRelease: User defined minor release number. Min=0,
               Max=4294967295, Default=0
build   User defined build number. Min=0,
                         Max=4294967295, Default=0
 Build: User defined build number. Min=0, Max=4294967295, Default=0
clearCacheDisabled   Clear cached secrets disabled.
 ClearCacheDisabled: Clear cached secrets disabled.

```
format                        Format of the image. Values are: 0 (VHD), 1
                              (VHDX). Default=0

    VHDX: If VHDX is true, the format of the image is VHDX. Otherwise it
                  is VHD. Default=false

deviceCount                   Read-only count of Devices. Default=0

    DeviceCount: Read-only count of Devices. Default=0

locked                        1 if the Disk is currently locked, 0
                              otherwise. Default=0

    Locked: True if the Disk is currently locked, false otherwise.
                  Default=false

records                       If successful, records retrieved.

    PvsDiskInfo[]: If successful, the PvsDiskInfo object(s) are returned.
```

Mcli-Get DiskInventory

```
Get the fields for Inventory Status of a Disk Version or all Disk Versions
   for a Disk Locator.
```

Get-PvsDiskInventory

```
Get the fields for Inventory Status of a Disk Version or all Disk Versions
   for a Disk Locator.

-p                            Parameters needed for the Get.

 This required

  diskLocatorId               GUID of the Disk Locator to Get Disk Version
                              Inventory of.

    Guid[] Guid or DiskLocatorId: GUID of the Disk Locator to Get Disk
                  Version Inventory of.

 or this required & resolution

  diskLocatorName             Name of the Disk Locator to Get Disk Version
                              Inventory of.

    string[] Name or DiskLocatorName: Name of the Disk Locator to Get
                  Disk Version Inventory of.

 Optional

 version                      Specific Version to Get.

   uint Version: Specific Version to Get.

 One of these resolutions when needed

 siteId                       GUID of the Site.

   Guid[] SiteId: GUID of the Site.

 siteName                     Name of the Site.

   string[] SiteName: Name of the Site.

 One of these resolutions when needed

 storeId                      GUID of the Store that is needed when a
                              diskLocatorName is used.
```

Guid[] StoreId: GUID of the Store that is needed when a
                    DiskLocatorName is used.

storeName   Name of the Store that is needed when a
                             diskLocatorName is used.

string[] StoreName: Name of the Store that is needed when a
                    DiskLocatorName is used.

Instead of a parameter that matches one of the members listed

PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

   (DiskLocatorId and Version) or DiskLocatorId

-f   List of fields to retrieve, if not included,
                             all fields are retrieved.

If only selected fields are needed, pass them in the Fields parameter
                    as a string array.

version   Version number. The base disk is version 0,
                             the other version numbers are in part of the
                             file name.

Version: Version number. The base disk is version 0, the other
                    version numbers are in part of the file name.

serverId   GUID of the Server that the Disk Version
                             Inventory is being reported about.

ServerId: GUID of the Server that the Disk Version Inventory is being
                    reported about.

serverName   Name of the Server that the Disk Version
                             Inventory is being reported about.

ServerName: Name of the Server that the Disk Version Inventory is
                    being reported about.

filePath   Path used to access the disk version from
                             the Server. Empty if the information is not
                             available.

FilePath: Path used to access the disk version from the Server. Empty
                    if the information is not available.

fileTime   Date/Time of the date version file. Format
                             is: YYYY-MM-DD HH:MM:SS. 0000-00-00 00:00:00
                             if the information is not available.

FileTime: Date/Time of the date version file. Has the date and time
                    without milliseconds. Empty if the information is
                    not available.

propertiesTime   Date/Time of the disk properties. Format is:
                             YYYY-MM-DD HH:MM:SS. 0000-00-00 00:00:00 if
                             the information is not available.

PropertiesTime: Date/Time of the disk properties. Has the date and
                    time without milliseconds. Empty if the information
                    is not available.

state   The number code of the inventory state.
                             Values are: 0 (Up to date), 1 (version file

is missing), 2 (version file is out of
date), 3 (properties are missing), 4
(properties are out of date), 5 (server is
not reachable).

State: The number code of the inventory state. Values are: 0 (Up to
date), 1 (version file is missing), 2 (version file
is out of date), 3 (properties are missing), 4
(properties are out of date), 5 (server is not
reachable).

active   1 if the Server is currently active, 2 if
                                unknown, and 0 otherwise.

Active: 1 if the Server is currently active, 2 if unknown, and 0
otherwise.

records   If successful, records retrieved.

PvsDiskInventory[]: If successful, the PvsDiskInventory object(s) are
returned.

Mcli-Get DiskLocator

Get the fields for a Disk Locator or all Disk Locators for a Device,
Server, Store, Site, or Farm. All DiskLocators are returned if no
parameters are passed.

Get-PvsDiskLocator

Get the fields for a Disk Locator or all Disk Locators for a Device,
Server, Store, Site, or Farm. All DiskLocators are returned if no
parameters are passed.

-p   Parameters needed for the Get.

 One of these optional

  diskLocatorId   GUID of the Disk Locator to Get.

   Guid[] Guid or DiskLocatorId: GUID of the Disk Locator to Get.

  deviceId   GUID of the Device to Get all DiskLocators
                                for.

   Guid[] DeviceId: GUID of the Device to Get all DiskLocators for.

  deviceName   Name of the Device to Get all DiskLocators
                                for.

   string[] DeviceName: Name of the Device to Get all DiskLocators for.

  deviceMac   MAC of the Device to Get all DiskLocators
                                for.

   PvsPhysicalAddress[] DeviceMac: MAC of the Device to Get all
                   DiskLocators for.

  serverId   GUID of the Server to Get all DiskLocators
                                for.

   Guid[] ServerId: GUID of the Server to Get all DiskLocators for.

  serverName   Name of the Server to Get all DiskLocators
                                for.

string[] ServerName: Name of the Server to Get all DiskLocators for.

  updateTaskId   GUID of the Update Task to Get all
                          DiskLocators for.

    Guid[] UpdateTaskId: GUID of the Update Task to Get all DiskLocators
                   for.

  siteId   GUID of the Site to Get all DiskLocators
                          for.

    Guid[] SiteId: GUID of the Site to Get all DiskLocators for.

  siteName   Name of the Site to Get all DiskLocators
                          for.

    string[] SiteName: Name of the Site to Get all DiskLocators for.

or one of these optional & resolutions

  diskLocatorName   Name of the Disk Locator File to Get.

    string[] Name or DiskLocatorName: Name of the Disk Locator File to
                   Get.

  storeId   GUID of the Store to Get all DiskLocators
                          for.

    Guid[] StoreId: GUID of the Store to Get all DiskLocators for.

  storeName   Name of the Store to Get all DiskLocators
                          for.

    string[] StoreName: Name of the Store to Get all DiskLocators for.

  updateTaskName   Name of the Update Task to Get all
                          DiskLocators for.

    string[] UpdateTaskName: Name of the Update Task to Get all
                   DiskLocators for.

One of these optional

  single   Include single server connection when set to
                          1. If this and All are not included, both
                          connection types are included.

    SwitchParameter Single: If -Single is specified, include single
                   server connection. If this and All are not included,
                   both connection types are included.

  all   Include all server connections for the store
                          when set to 1. If this and Single are not
                          included, both connection types are
                          included.

    SwitchParameter All: If -All is specified, include all server
                   connections for the store. If this and Single are
                   not included, both connection types are included.

Optional

  onlyActive   Include only the active DiskLocators when
                          set to 1. If not included or set to 0, all
                          DiskLocators are returned.

updateDevice   Include DiskLocators that have an Update Device when set to 1. Include DiskLocators that do not have an Update Device when set to 0. If this parameter is not included, then all DiskLocators are returned.

One of these resolutions when needed

siteId   GUID of the Site.

siteName   Name of the Site.

One of these resolutions when needed

storeId   GUID of the Store that is needed when a diskLocatorName is used.

storeName   Name of the Store that is needed when a diskLocatorName is used.

Instead of a parameter that matches one of the members listed

-s   Name of field to sort on. If not included, the first field listed is used. Append a - to the field name to get descending order.

diskLocatorName   Name of the Disk Locator File. It is unique within the Store. ASCII Max Length=52

description   User description. Default="" Max Length=250

menuText   Text that is displayed in the Boot Menu. If this field has no value, the name value is used. Default="" ASCII Max Length=64

-f   List of fields to retrieve, if not included, all fields are retrieved.

diskLocatorId   Read-only GUID that uniquely identifies this Disk Locator. Length=36

diskLocatorName   Name of the Disk Locator File. It is unique
                         within the Store. ASCII Max Length=52

 Name or DiskLocatorName: Name of the Disk Locator File. It is unique
                 within the Store. ASCII Max Length=52

siteId   GUID of the Site this DiskLocator is to be a
                         member of. It is not used with siteName.
                         Length=36

 SiteId: GUID of the Site this DiskLocator is to be a member of. It is
                 not used with SiteName.

siteName   Name of the Site this DiskLocator is to be a
                         member of. It is not used with siteId.

 SiteName: Name of the Site this DiskLocator is to be a member of. It
                 is not used with SiteId.

storeId   GUID of the Store that this Disk Locator is
                         a member of. siteName or siteId must also be
                         used. It is not used with storeName.
                         Length=36

 StoreId: GUID of the Store that this Disk Locator is a member of.
                 SiteName or SiteId must also be used. It is not used
                 with StoreName.

storeName   Name of the Store that this Disk Locator is
                         a member of. siteName or siteId must also be
                         used. It is not used with storeId.

 StoreName: Name of the Store that this Disk Locator is a member of.
                 SiteName or SiteId must also be used. It is not used
                 with StoreId.

description   User description. Default="" Max Length=250

 Description: User description. Default="" Max Length=250

menuText   Text that is displayed in the Boot Menu. If
                         this field has no value, the name value is
                         used. Default="" ASCII Max Length=64

 MenuText: Text that is displayed in the Boot Menu. If this field has
                 no value, the name value is used. Default="" ASCII
                 Max Length=64

serverId   GUID of the single Server that this Disk
                         Locator is assigned to. It is not used with
                         serverName. Default=null Length=36

 ServerId: GUID of the single Server that this Disk Locator is
                 assigned to. It is not used with ServerName.
                 Default=00000000-0000-0000-0000-000000000000

serverName   Name of the single Server that this Disk
                         Locator is assigned to. It is not used with
                         serverId. Default=null

 ServerName: Name of the single Server that this Disk Locator is
                 assigned to. It is not used with ServerId.
                 Default=""

enabled	1 when this disk can be booted, 0 otherwise. Default=1

Enabled: True when this disk can be booted, false otherwise. Default=true

role	Read-only Role of the user for this item. 100 is Farm Administrator, 200 is Site Administrator, 300 is Collection Administrator, and 999 is read-only. Default=999

Role: Read-only Role of the user for this item. 100 is Farm Administrator, 200 is Site Administrator, 300 is Collection Administrator, and 999 is read-only. Default=999

mapped	1 if the Disk is currently mapped, 0 otherwise. Default=0

Mapped: True if the Disk is currently mapped, false otherwise. Default=false

enabledForDevice	1 when this disk is enabled for the Device specified, 0 otherwise. This is only returned when a Device is specified. Default=1

EnabledForDevice: True when this disk is enabled for the Device specified, false otherwise. This is only returned when a Device is specified. Default=true

active	1 if the DiskLocator is currently active, 0 otherwise. Default=0

Active: True if the DiskLocator is currently active, false otherwise. Default=false

rebalanceEnabled	1 when this Server can automatically rebalance Devices, 0 otherwise. Default=0

RebalanceEnabled: True when this Server can automatically rebalance Devices, false otherwise. Default=false

rebalanceTriggerPercent	Percent over fair load that triggers a dynamic Device rebalance. Min=5, Max=5000, Default=25

RebalanceTriggerPercent: Percent over fair load that triggers a dynamic Device rebalance. Min=5, Max=5000, Default=25

subnetAffinity	Qualifier for subnet affinity when assigning a Server. 0=None, 1=Best Effort, 2=Fixed. Min=0, Max=2, Default=0

SubnetAffinity: Qualifier for subnet affinity when assigning a Server. 0=None, 1=Best Effort, 2=Fixed. Min=0, Max=2, Default=0

diskUpdateDeviceId	GUID of the DiskUpdateDevice that is used when updates are performed. Default=null

```
            DiskUpdateDeviceId: GUID of the DiskUpdateDevice that is used when
                      updates are performed. Default=00000000-0000-0000-
                      0000-000000000000
        diskUpdateDeviceName        Name of the DiskUpdateDevice that is used
                                    when updates are performed. Default=null

            DiskUpdateDeviceName: Name of the DiskUpdateDevice that is used when
                      updates are performed. Default=""
        temporaryVersionSet         Read-only 1 when temporary version(s) are
                                    set. Default=0

            TemporaryVersionSet: Read-only true when temporary version(s) are
                      set. Default=false
    records                         If successful, records retrieved.

        PvsDiskLocator[]: If successful, the PvsDiskLocator object(s) are
                      returned.
```

Mcli-Get DiskLocatorLock

```
    Get the fields for all the locks of a Disk Locator.
```

Get-PvsDiskLocatorLock

```
    Get the fields for all the locks of a Disk Locator.
    -p                              Parameters needed for the Get.
     This required
      diskLocatorId                 GUID of the Disk Locator to Get the Locks.

        Guid[] Guid or DiskLocatorId: GUID of the Disk Locator to Get the
                      Locks.
     or this required & resolution
      diskLocatorName               Name of Disk Locator to Get the Locks.

        string[] Name or DiskLocatorName: Name of Disk Locator to Get the
                      Locks.
    One of these resolutions when needed
     siteId                         GUID of the Site.

        Guid[] SiteId: GUID of the Site.
     siteName                       Name of the Site.

        string[] SiteName: Name of the Site.
    One of these resolutions when needed
     storeId                        GUID of the Store that is needed when a
                                    diskLocatorName is used.

        Guid[] StoreId: GUID of the Store that is needed when a
                      DiskLocatorName is used.
     storeName                      Name of the Store that is needed when a
                                    diskLocatorName is used.

        string[] StoreName: Name of the Store that is needed when a
                      DiskLocatorName is used.
```

```
        Instead of a parameter that matches one of the members listed
```

PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

    DiskLocatorId

```
   -f                              List of fields to retrieve, if not included,
                                   all fields are retrieved.
```

If only selected fields are needed, pass them in the Fields parameter
                    as a string array.

```
      exclusive                    1 when the lock is exclusive, 0 when it is
                                   shared. Default=0
```

Exclusive: True when the lock is exclusive, false when it is shared.
                    Default=false

```
      deviceId                     GUID of the Device that has the lock, will
                                   not be included if a Server has the lock.
```

DeviceId: GUID of the Device that has the lock, will be 00000000-
                    0000-0000-0000-000000000000 if a Server has the
                    lock.

```
      deviceName                   Name of the Device that has the lock, will
                                   not be included if a Server has the lock.
```

DeviceName: Name of the Device that has the lock, will not be
                    included if a Server has the lock.

```
      serverId                     GUID of the Server that has the lock, will
                                   not be included if a Device has the lock.
```

ServerId: GUID of the Server that has the lock, will be 00000000-
                    0000-0000-0000-000000000000 if a Device has the
                    lock.

```
      serverName                   Name of the Server that has the lock, will
                                   not be included if a Device has the lock.
```

ServerName: Name of the Server that has the lock, will not be
                    included if a Device has the lock.

```
      readOnly                     1 when lock is because file system is read
                                   only, 0 when file system is read write
                                   Default=0
```

ReadOnly: True when lock is because file system is read only, false
                    when file system is read write Default=false

```
   records                         If successful, records retrieved.
```

PvsDiskLocatorLock[]: If successful, the PvsDiskLocatorLock object(s)
                    are returned.

Mcli-Get DiskUpdateDevice

```
   Get the fields and status for a Disk Update Device, or all Disk Update
      Devices for a Site, Server, DiskLocator or Farm. All Disk Update Devices
      are returned if no parameters are passed.
```

Get-PvsDiskUpdateDevice

Get the fields and status for a Disk Update Device, or all Disk Update
    Devices for a Site, Server, DiskLocator or Farm. All Disk Update Devices
    are returned if no parameters are passed.

    -p   Parameters needed for the Get.

   One of these optional

    deviceId   GUID of the Disk Update Device to Get.

     Guid[] Guid or DeviceId: GUID of the Disk Update Device to Get.

    deviceName   Name of Disk Update Device to Get.

     string[] Name or DeviceName: Name of Disk Update Device to Get.

    deviceMac   MAC of the Disk Update Device to Get.

     PvsPhysicalAddress[] DeviceMac: MAC of the Disk Update Device to Get.

    serverId   GUID of the Server to Get all Disk Update
                                    Devices for.

     Guid[] ServerId: GUID of the Server to Get all Disk Update Devices
                        for.

    serverName   Name of the Server to Get all Disk Update
                                    Devices for.

     string[] ServerName: Name of the Server to Get all Disk Update
                      Devices for.

    diskLocatorId   GUID of the DiskLocator to Get the Disk
                                    Update Device for.

     Guid[] DiskLocatorId: GUID of the DiskLocator to Get the Disk Update
                      Device for.

    updateTaskId   GUID of the Update Task to Get all Disk
                                    Update Devices for.

     Guid[] UpdateTaskId: GUID of the Update Task to Get all Disk Update
                      Devices for.

    siteId   GUID of the Site.

     Guid[] SiteId: GUID of the Site.

    siteName   Name of the Site.

     string[] SiteName: Name of the Site.

   or one of these optional & resolutions

    diskLocatorName   Name of the DiskLocator to Get the Disk
                                    Update Device for.

     string[] DiskLocatorName: Name of the DiskLocator to Get the Disk
                      Update Device for.

    updateTaskName   Name of the Update Task to Get all Disk
                                    Update Devices for.

     string[] UpdateTaskName: Name of the Update Task to Get all Disk
                      Update Devices for.

   Optional

onlyActive   Include only the active Disk Update Devices
                                                when set to 1. If not included or set to 0,
                                                all Disk Update Devices are returned. Only
                                                active Disk Update Devices are always
                                                returned for serverId or serverName.

        SwitchParameter OnlyActive: If -OnlyActive is specified, include only
                        the active Disk Update Devices. Only active Disk
                        Update Devices are always returned for ServerId or
                        ServerName.

                makLicenseActivated   Optional MAK licensing indicator value to
                                                only return active Disk Update Devices for.
                                                Values are: 0 (MAK not used), 1 (Not
                                                Activated), 2 (Activated).

            uint MakLicenseActivated: Optional MAK licensing indicator value to
                        only return active Disk Update Devices for. Values
                        are: 0 (MAK not used), 1 (Not Activated), 2
                        (Activated).

        One of these resolutions when needed

            siteId   GUID of the Site.

                Guid[] SiteId: GUID of the Site.

            siteName   Name of the Site.

                string[] SiteName: Name of the Site.

        One of these resolutions when needed

            storeId   GUID of the Store that is needed when a
                                                diskLocatorName is used.

                Guid[] StoreId: GUID of the Store that is needed when a
                        DiskLocatorName is used.

            storeName   Name of the Store that is needed when a
                                                diskLocatorName is used.

                string[] StoreName: Name of the Store that is needed when a
                        DiskLocatorName is used.

        Instead of a parameter that matches one of the members listed

            PvsObject[] Object: PvsObjects with the members below can be used as
                        the Object parameter or from a pileline:

                DeviceId, ServerId, DiskLocatorId, UpdateTaskId or SiteId

        -s   Name of field to sort on. If not included,
                                                the first field listed is used. Append a - to
                                                the field name to get descending order.

            deviceName   Computer name with no spaces. ASCII computer
                                                name characters Max Length=15

            virtualHostingPoolName   Name of the Virtual Hosting Pool.

            diskLocatorName   Name of the Disk Locator File to update with
                                                this Device.

            description   User description. Default="" Max Length=250

deviceMac   Ethernet address as a string in the form XX-XX-XX-XX-XX-XX. Uniquely identifies the Device. Length=17

port   UDP port to use with Stream Service. Min=1025, Max=65534, Default=6901

-f   List of fields to retrieve, if not included, all fields are retrieved.

If only selected fields are needed, pass them in the Fields parameter as a string array.

deviceId   Read-only GUID that uniquely identifies this Device. Length=36

Guid or DeviceId: Read-only GUID that uniquely identifies this Device.

deviceName   Computer name with no spaces. ASCII computer name characters Max Length=15

Name or DeviceName: Computer name with no spaces. ASCII computer name characters Max Length=15

virtualHostingPoolId   GUID of the Virtual Hosting Pool. It is not used with virtualHostingPoolName. Default=null Length=36

VirtualHostingPoolId: GUID of the Virtual Hosting Pool. It is not used with VirtualHostingPoolName. Default=00000000-0000-0000-0000-000000000000

virtualHostingPoolName   Name of the Virtual Hosting Pool.

VirtualHostingPoolName: Name of the Virtual Hosting Pool.

diskLocatorId   GUID of the Disk Locator to update with this Device.

DiskLocatorId: GUID of the Disk Locator to update with this Device.

diskLocatorName   Name of the Disk Locator File to update with this Device.

DiskLocatorName: Name of the Disk Locator File to update with this Device.

siteId   GUID of the Site this Disk Update Device is to be a member of.

SiteId: GUID of the Site this Disk Update Device is to be a member of.

siteName   Name of the Site this Disk Update Device is to be a member of.

SiteName: Name of the Site this Disk Update Device is to be a member of.

storeId   GUID of the Store that the Disk Locator is a member of.

StoreId: GUID of the Store that the Disk Locator is a member of.

storeName   Name of the Store that the Disk Locator is a member of.

StoreName: Name of the Store that the Disk Locator is a member of.

description   User description. Default="" Max Length=250

Description: User description. Default="" Max Length=250

deviceMac   Ethernet address as a string in the form XX-
                        XX-XX-XX-XX-XX. Uniquely identifies the
                        Device. Length=17

DeviceMac: Ethernet address can have the form XX-XX-XX-XX-XX-XX.
           Uniquely identifies the Device.

port   UDP port to use with Stream Service.
                        Min=1025, Max=65534, Default=6901

Port: UDP port to use with Stream Service. Min=1025, Max=65534,
      Default=6901

active   1 if the Device is currently active, 0
                        otherwise. Default=0

Active: True if the Device is currently active, false otherwise.
        Default=false

adTimestamp   The time the Active Directory machine
                        account password was generated. Do not set
                        this field, it is only set internally by
                        PVS. Default=0

AdTimestamp: The time the Active Directory machine account password
             was generated. Do not set this field, it is only set
             internally by PVS. Default=0

adSignature   The signature of the Active Directory
                        machine account password. Do not set this
                        field, it is only set internally by PVS.
                        Default=0

AdSignature: The signature of the Active Directory machine account
             password. Do not set this field, it is only set
             internally by PVS. Default=0

logLevel   Level to perform logging at. Values are: 0
                        (None), 1 (Fatal), 2 (Error), 3 (Warning), 4
                        (Info), 5 (Debug), and 6 (Trace). Min=0,
                        Max=6, Default=0

LogLevel: Level to perform logging at. Values are: 0 (None), 1
          (Fatal), 2 (Error), 3 (Warning), 4 (Info), 5
          (Debug), and 6 (Trace). Min=0, Max=6, Default=0

domainName   Fully qualified name of the domain that the
                        Device belongs to. Do not set this field, it
                        is only set internally by PVS. Default=null
                        Max Length=255

DomainName: Fully qualified name of the domain that the Device
            belongs to. Do not set this field, it is only set
            internally by PVS. Default="" Max Length=255

domainObjectSID   The value of the objectSID AD attribute of
                        the same name for the Device's computer
                        account. Do not set this field, it is only

set internally by PVS. Default=null Max Length=186

DomainObjectSID: The value of the objectSID AD attribute of the same name for the Device's computer account. Do not set this field, it is only set internally by PVS. Default="" Max Length=186

domainControllerName   The name of the DC used to create the host's computer account. Do not set this field, it is only set internally by PVS. Default=null Max Length=4000

DomainControllerName: The name of the DC used to create the host's computer account. Do not set this field, it is only set internally by PVS. Default="" Max Length=4000

domainTimeCreated   The time that the computer account was created. Do not set this field, it is only set internally by PVS. Format is: YYYY-MM-DD HH:MM:SS.SSS Default=null

DomainTimeCreated: The time that the computer account was created. Has the date and time including milliseconds. Do not set this field, it is only set internally by PVS. Default=Empty

adPassword   The Active Directory machine account password. Do not set this field, it is only set internally by PVS. Default="" Max Length=256

AdPassword: The Active Directory machine account password. Do not set this field, it is only set internally by PVS. Default="" Max Length=256

ip   Read-only IP of the Device. It is equal to "" if the Device is not active.

Ip: Read-only IP of the Device. It is equal to 0.0.0.0 if the Device is not active.

serverPortConnection   Read-only Port of the Server that the Device is using. It is equal to "" if the Device is not active. Default=0

ServerPortConnection: Read-only Port of the Server that the Device is using. It is equal to 0 if the Device is not active. Default=0

serverIpConnection   Read-only IP of the Server that the Device is using. It is equal to "" if the Device is not active.

ServerIpConnection: Read-only IP of the Server that the Device is using. It is equal to 0.0.0.0 if the Device is not active.

serverId   Read-only GUID of the Server that the Device is using. It is equal to "" if the Device is not active. Length=36

ServerId: Read-only GUID of the Server that the Device is using. It is equal to 00000000-0000-0000-0000-000000000000 if the Device is not active.

serverName   Read-only Name of the Server that the Device is using. It is equal to "" if the Device is not active.

ServerName: Read-only Name of the Server that the Device is using. It is equal to "" if the Device is not active.

diskVersion   Read-only version of the Disk Locator File that the Device is using. It is equal to "" if the Device is not active. Default=0

DiskVersion: Read-only version of the Disk Locator File that the Device is using. It is equal to 0 if the Device is not active. Default=0

status   1 or 2 numbers in the format n,n. They are the number of retries and if ram cache is being used, ram cache percent used. It is equal to "" if the Device is not active.

Status: 1 or 2 numbers in the format n,n. They are the number of retries and if ram cache is being used, ram cache percent used. It is equal to "" if the Device is not active.

licenseType   0 when None, 1 for Desktop, 2 for Server, 5 for OEM SmartClient, 6 for XenApp, 7 for XenDesktop. It is equal to 0 if the Device is not active. Default=0

LicenseType: 0 when None, 1 for Desktop, 2 for Server, 5 for OEM SmartClient, 6 for XenApp, 7 for XenDesktop. It is equal to 0 if the Device is not active. Default=0

makLicenseActivated   Read-only indicator if MAK licensing is being used and is activated. Values are: 0 (MAK not used), 1 (Not Activated), 2 (Activated). It is equal to "" if the Device is not active. Default=0

MakLicenseActivated: Read-only indicator if MAK licensing is being used and is activated. Values are: 0 (MAK not used), 1 (Not Activated), 2 (Activated). It is equal to 0 if the Device is not active. Default=0

model   Oem Only: Read-only model of the computer. Values are OptiPlex 745, 755, 320, 760, FX160, or Default. It is equal to "" if the Device is not active.

Model: Oem Only: Read-only model of the computer. Values are OptiPlex 745, 755, 320, 760, FX160, or Default. It is equal to "" if the Device is not active.

license   Oem Only: Read-only type of the license. Values are 0 when None, 1 or 2 when Desktop. It is equal to 0 if the Device is not active. Default=0

records   If successful, records retrieved.

Mcli-Get DiskUpdateStatus

Get the status of an Update Task, or all Update Tasks for a Site or Farm.
All Disk Update Tasks are returned if no parameters are passed.

Get-PvsDiskUpdateStatus

Get the status of an Update Task, or all Update Tasks for a Site or Farm.
All Disk Update Tasks are returned if no parameters are passed.

-p   Parameters needed for the Get.

One of these optional

updateTaskId   GUID of the Update Task to Get.

Guid[] UpdateTaskId: GUID of the Update Task to Get.

deviceId   GUID of the Disk Update Device to Get Disk
Update Status for.

Guid[] DeviceId: GUID of the Disk Update Device to Get Disk Update
Status for.

deviceName   Name of the Disk Update Device to Get Disk
Update Status for.

string[] DeviceName: Name of the Disk Update Device to Get Disk
Update Status for.

deviceMac   MAC of the Disk Update Device to Get Disk
Update Status for.

PvsPhysicalAddress[] DeviceMac: MAC of the Disk Update Device to Get
Disk Update Status for.

diskUpdateTaskId   GUID of the Disk Update Task and Device
relationship to Get Disk Update Status for.

Guid[] Guid or DiskUpdateTaskId: GUID of the Disk Update Task and
Device relationship to Get Disk Update Status for.

siteId   GUID of the Site to Get all Update Tasks
for. Also used with updateTaskName.

Guid[] SiteId: GUID of the Site to Get all Update Tasks for. Also
used with UpdateTaskName.

siteName   Name of the Site to Get all Update Tasks
for. Also used with updateTaskName.

string[] SiteName: Name of the Site to Get all Update Tasks for. Also
used with UpdateTaskName.

or this optional & resolution

updateTaskName   Name of the Update Task to Get.

string[] UpdateTaskName: Name of the Update Task to Get.

One of these resolutions when needed

siteId   GUID of the Site.

   Guid[] SiteId: GUID of the Site to Get all Update Tasks for. Also
             used with UpdateTaskName.

siteName   Name of the Site.

   string[] SiteName: Name of the Site to Get all Update Tasks for. Also
             used with UpdateTaskName.

Instead of a parameter that matches one of the members listed

   PvsObject[] Object: PvsObjects with the members below can be used as
             the Object parameter or from a pileline:

    UpdateTaskId, DeviceId or SiteId

-s   Name of field to sort on. If not included,
                                the first field listed is used. Append a - to
                                the field name to get descending order.

updateTaskName   Name of the Update Task.

description   User description of the Update Task.

diskLocatorName   Name of the Disk Locator File to update.

virtualHostingPoolName   Name of the Virtual Hosting Pool being used
                                 for the update.

deviceName   Name of the Device being used to do the
                                 update.

-f   List of fields to retrieve, if not included,
                                all fields are retrieved.

  If only selected fields are needed, pass them in the Fields parameter
             as a string array.

diskUpdateTaskId   GUID that uniquely identifies this Update
                                 Task and Device relationship. Length=36

  Guid or DiskUpdateTaskId: GUID that uniquely identifies this Update
           Task and Device relationship.

updateTaskId   GUID that uniquely identifies the Update
                                 Task. Length=36

  UpdateTaskId: GUID that uniquely identifies the Update Task.

updateTaskName   Name of the Update Task.

  UpdateTaskName: Name of the Update Task.

description   User description of the Update Task.

  Description: User description of the Update Task.

diskLocatorId   GUID of the Disk Locator to update.

  DiskLocatorId: GUID of the Disk Locator to update.

diskLocatorName   Name of the Disk Locator File to update.

  Name or DiskLocatorName: Name of the Disk Locator File to update.

virtualHostingPoolId   GUID of the Virtual Hosting Pool being used
                           for the update.

  VirtualHostingPoolId: GUID of the Virtual Hosting Pool being used for
                   the update.

virtualHostingPoolName   Name of the Virtual Hosting Pool being used
                           for the update.

  VirtualHostingPoolName: Name of the Virtual Hosting Pool being used
                   for the update.

deviceId   GUID that Device being used to do the
                           update.

  DeviceId: GUID that Device being used to do the update.

deviceName   Name of the Device being used to do the
                           update.

  DeviceName: Name of the Device being used to do the update.

siteId   GUID of the Site that this Update Task Name
                           is a member of.

  SiteId: GUID of the Site that this Update Task Name is a member of.

siteName   Name of the Site that this Update Task Name
                           is a member of.

  SiteName: Name of the Site that this Update Task Name is a member of.

storeId   GUID of the Store that the Disk Locator is a
                           member of.

  StoreId: GUID of the Store that the Disk Locator is a member of.

storeName   Name of the Store that the Disk Locator is a
                           member of.

  StoreName: Name of the Store that the Disk Locator is a member of.

previousResult   Status of the last run. Values are: 0
                           (Ready), 1 (Update Pending), 2 (Preparing
                           Image), 3 (Starting VM), 4 (Update In
                           Progress), 5 (Stopping VM), 6 (Submitting
                           Image), 7 (Reverting Image), 8 (Invalid), 9
                           (Aborted), 10 (Completed Successfully), 11
                           (No Updates) Min=0, Max=11, Default=0

  PreviousResult: Status of the last run. Values are: 0 (Ready), 1
                   (Update Pending), 2 (Preparing Image), 3 (Starting
                   VM), 4 (Update In Progress), 5 (Stopping VM), 6
                   (Submitting Image), 7 (Reverting Image), 8
                   (Invalid), 9 (Aborted), 10 (Completed Successfully),
                   11 (No Updates) Min=0, Max=11, Default=0

previousResultMessage   Message string that includes the results of
                           the last run. Default="" Max Length=255

  PreviousResultMessage: Message string that includes the results of
                   the last run. Default="" Max Length=255

currentStatus   Current status of the update. Values are: 0
                           (Ready), 1 (Update Pending), 2 (Preparing
                           Image), 3 (Starting VM), 4 (Update In

```
                             Progress), 5 (Stopping VM), 6 (Submitting
                             Image), 7 (Reverting Image), 8 (Invalid), 9
                             (Aborted), 10 (Completed Successfully), 11
                             (No Updates) Min=0, Max=11, Default=0
```

CurrentStatus: Current status of the update. Values are: 0 (Ready), 1 (Update Pending), 2 (Preparing Image), 3 (Starting VM), 4 (Update In Progress), 5 (Stopping VM), 6 (Submitting Image), 7 (Reverting Image), 8 (Invalid), 9 (Aborted), 10 (Completed Successfully), 11 (No Updates) Min=0, Max=11, Default=0

```
     currentStatusMessage         Message string that includes the results of
                                  the run. Default="" Max Length=255
```

CurrentStatusMessage: Message string that includes the results of the run. Default="" Max Length=255

```
   records                        If successful, records retrieved.
```

PvsDiskUpdateStatus[]: If successful, the PvsDiskUpdateStatus object(s) are returned.

Mcli-Get DiskVersion

Get the fields for a Disk Version or all Disk Versions for a Disk Locator.

Get-PvsDiskVersion

Get the fields for a Disk Version or all Disk Versions for a Disk Locator.

```
     -p                           Parameters needed for the Get.

    This required

    diskLocatorId                 GUID of the Disk Locator to Get Disk
                                  Versions of.
```

Guid[] Guid or DiskLocatorId: GUID of the Disk Locator to Get Disk Versions of.

```
   or this required & resolution

    diskLocatorName               Name of the Disk Locator to Get Disk
                                  Versions of.
```

string[] Name or DiskLocatorName: Name of the Disk Locator to Get Disk Versions of.

```
   One of these optional

    version                       Specific Version to Get.
```

uint Version: Specific Version to Get.

```
    type                          When set to 1, get the Maintenance or
                                  MaintenanceHighestVersion access version if
                                  it exists. When set to 2, get the Test
                                  access versions if any exist. When set to 3,
                                  get the Override access version if it
                                  exists.
```

uint Type: When set to 1, get the Maintenance or MaintenanceHighestVersion access version if it exists. When set to 2, get the Test access versions

if any exist. When set to 3, get the Override access
                              version if it exists.

One of these resolutions when needed

  siteId   GUID of the Site.

    Guid[] SiteId: GUID of the Site.

  siteName   Name of the Site.

    string[] SiteName: Name of the Site.

One of these resolutions when needed

  storeId   GUID of the Store that is needed when a
                              diskLocatorName is used.

    Guid[] StoreId: GUID of the Store that is needed when a
                    DiskLocatorName is used.

  storeName   Name of the Store that is needed when a
                              diskLocatorName is used.

    string[] StoreName: Name of the Store that is needed when a
                    DiskLocatorName is used.

  Instead of a parameter that matches one of the members listed

    PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

      (DiskLocatorId and Version) or DiskLocatorId

-s   Name of field to sort on. If not included,
                              the first field listed is used. Append a - to
                              the field name to get descending order.

  version   Read-only version number. The base disk is
                              version 0, the other version numbers are in
                              part of the file name. Default=0

-f   List of fields to retrieve, if not included,
                              all fields are retrieved.

  If only selected fields are needed, pass them in the Fields parameter
                  as a string array.

  version   Read-only version number. The base disk is
                              version 0, the other version numbers are in
                              part of the file name. Default=0

    Version: Read-only version number. The base disk is version 0, the
                    other version numbers are in part of the file name.
                    Default=0

  description   User description. Default="" Max Length=250

    Description: User description. Default="" Max Length=250

  type   Read-only type of the Disk Version. Values
                              are: 0 (Base), 1 (Manual), 2 (Automatic), 3
                              (Merge), and 4 (MergeBase) Min=0, Max=4,
                              Default=0

    Type: Read-only type of the Disk Version. Values are: 0 (Base), 1
                    (Manual), 2 (Automatic), 3 (Merge), and 4
                    (MergeBase) Min=0, Max=4, Default=0

createDate   Read-only Date/Time that the Disk Version
                           was created. Default=getdate

*CreateDate: Read-only Date/Time that the Disk Version was created.
                Default=getdate*

scheduledDate   Date/Time that the Disk Version is scheduled
                           to become available. Format is: YYYY-MM-DD
                           HH:MM. When empty, the disk version is made
                           available immediately. Default=null

*ScheduledDate: Date/Time that the Disk Version is scheduled to become
                available. Has the date, hour and minute. Empty when
                the disk version is made available immediately.
                Default=Empty*

deleteWhenFree   Read-only 1 if the Disk Version is no longer
                           needed because of a merge. If not current
                           booted by a Device, it can be deleted.
                           Default=0

*DeleteWhenFree: Read-only true if the Disk Version is no longer
                needed because of a merge. If not current booted by
                a Device, it can be deleted. Default=false*

access   Read-only access of the Disk Version. Values
                           are: 0 (Production), 1 (Maintenance), 2
                           (MaintenanceHighestVersion), 3 (Override), 4
                           (Merge), 5 (MergeMaintenance), 6
                           (MergeTest), and 7 (Test) Min=0, Max=7,
                           Default=0

*Access: Read-only access of the Disk Version. Values are: 0
                (Production), 1 (Maintenance), 2
                (MaintenanceHighestVersion), 3 (Override), 4
                (Merge), 5 (MergeMaintenance), 6 (MergeTest), and 7
                (Test) Min=0, Max=7, Default=0*

diskFileName   Name of the Disk File including the
                           extension. Default=null

*Name or DiskFileName: Name of the Disk File including the extension.
                Default=""*

deviceCount   Read-only count of Devices. Default=0

DeviceCount: Read-only count of Devices. Default=0

goodInventoryStatus   1 when the up to date file is accessible by
                           all Servers, 0 otherwise. Default=0

*GoodInventoryStatus: True when the up to date file is accessible by
                all Servers, false otherwise. Default=false*

taskId   When a Merge is occurring, this will be set
                           with the task number of the process that is
                           occurring. Default=null

*TaskId: When a Merge is occurring, this will be set with the task
                number of the process that is occurring. Default=""*

canDelete   Read-only 1 when the version can be deleted.
                           Default=0

CanDelete: Read-only true when the version can be deleted. Default=false

canMerge   Read-only 1 when the version can be update merged. Will be set for the highest version number. Default=0

CanMerge: Read-only true when the version can be update merged. Will be set for the highest version number. Default=false

canMergeBase   Read-only 1 when the version can be base merged. Will be set for the highest version number. Default=0

CanMergeBase: Read-only true when the version can be base merged. Will be set for the highest version number. Default=false

canPromote   Read-only 1 when the version can be promoted. Default=0

CanPromote: Read-only true when the version can be promoted. Default=false

canRevertTest   Read-only 1 when the version can be reverted to Test Access. Default=0

CanRevertTest: Read-only true when the version can be reverted to Test Access. Default=false

canRevertMaintenance   Read-only 1 when the version can be reverted to Maintenance Access. Default=0

CanRevertMaintenance: Read-only true when the version can be reverted to Maintenance Access. Default=false

canSetScheduledDate   Read-only 1 when the version can have the scheduled date modified. Default=0

CanSetScheduledDate: Read-only true when the version can have the scheduled date modified. Default=false

canOverride   Read-only 1 when the version can be set as the Override. Default=0

CanOverride: Read-only true when the version can be set as the Override. Default=false

isPending   Read-only 1 when the version scheduledDate has not occurred. Default=0

IsPending: Read-only true when the version ScheduledDate has not occurred. Default=false

temporaryVersionSet   Read-only 1 when temporary version(s) are set. Some changes cannot be made to the version when this is set. Default=0

TemporaryVersionSet: Read-only true when temporary version(s) are set. Some changes cannot be made to the version when this is set. Default=false

records   If successful, records retrieved.

PvsDiskVersion[]: If successful, the PvsDiskVersion object(s) are returned.

Mcli-Get Farm

Get the fields for the Farm.

Get-PvsFarm

Get the fields for the Farm.

   -p   Parameters needed for the Get.

   Optional

     farmId   GUID of the Farm to Get. This is optional since there is only one Farm.

       Guid[] Guid or FarmId: GUID of the Farm to Get. This is optional since there is only one Farm.

    Instead of a parameter that matches one of the members listed

      PvsObject[] Object: PvsObjects with the members below can be used as the Object parameter or from a pileline:

       FarmId

  -f   List of fields to retrieve, if not included, all fields are retrieved.

   If only selected fields are needed, pass them in the Fields parameter as a string array.

     farmId   Read-only GUID that uniquely identifies this Farm. Length=36

      Guid or FarmId: Read-only GUID that uniquely identifies this Farm.

     farmName   Name of the Farm. Default="" Max Length=50

      Name or FarmName: Name of the Farm. Default="" Max Length=50

     description   User description. Default="" Max Length=250

      Description: User description. Default="" Max Length=250

     autoAddEnabled   1 when Auto Add is enabled, 0 otherwise. Default=0

      AutoAddEnabled: True when Auto Add is enabled, false otherwise. Default=false

     auditingEnabled   1 when Auditing is enabled, 0 otherwise. Default=0

      AuditingEnabled: True when Auditing is enabled, false otherwise. Default=false

     lastAuditArchiveDate Last date of Audit Trail data that was Archived. Format is yyyy/mm/dd Default=null

      LastAuditArchiveDate: Last date of Audit Trail data that was Archived. Has the date. Default=Empty

     defaultSiteId   GUID of the Site to place new Devices into automatically. Not used with defaultSiteName. Default=null Length=36

DefaultSiteId: GUID of the Site to place new Devices into
automatically. Not used with defaultSiteName.
Default=00000000-0000-0000-0000-000000000000

defaultSiteName   Name of the Site to place new Devices into
                         automatically. Not used with defaultSiteId.
                         Default=null

DefaultSiteName: Name of the Site to place new Devices into
automatically. Not used with DefaultSiteId.
Default=""

offlineDatabaseSupportEnabled 1 when Offline Database Support is
                         enabled, 0 otherwise. Default=0

OfflineDatabaseSupportEnabled: True when Offline Database Support is
enabled, false otherwise. Default=false

adGroupsEnabled   Active Directory groups are used for
                         authorization, when set to 1. Windows groups
                         are used when set to 0. Default=0

AdGroupsEnabled: Active Directory groups are used for authorization,
when set to true. Windows groups are used when set
to false. Default=false

licenseServer   License server name. Default="" Max
                         Length=255

LicenseServer: License server name. Default="" Max Length=255

licenseServerPort   License server port. Min=1025, Max=65534,
                         Default=27000

LicenseServerPort: License server port. Min=1025, Max=65534,
Default=27000

licenseTradeUp   License server trade up, when set to 1.
                         Default=0

LicenseTradeUp: License server trade up, when set to true.
Default=false

automaticMergeEnabled   1 when Automatic Merge is enabled, 0
                         otherwise. If the number of versions becomes
                         more than the maxVersions value, a merge
                         will occur at the end of PromoteDiskVersion.
                         Default=1

AutomaticMergeEnabled: True when Automatic Merge is enabled, false
otherwise. If the number of versions becomes more
than the MaxVersions value, a merge will occur at
the end of PromoteDiskVersion. Default=true

maxVersions   Maximum number a versions of a Disk that can
                         exist before a merge will automatically
                         occur. Min=3, Max=50, Default=5

MaxVersions: Maximum number a versions of a Disk that can exist
before a merge will automatically occur. Min=3,
Max=50, Default=5

mergeMode   Mode to place the version in after a merge
                         has occurred. Values are: 0 (Production), 1

```
                                      (Test) and 2 (Maintenance). Min=0, Max=2,
                                      Default=2

        MergeMode: Mode to place the version in after a merge has occurred.
                         Values are: 0 (Production), 1 (Test) and 2
                         (Maintenance). Min=0, Max=2, Default=2

    databaseServerName          Read-only name of the database server.

     DatabaseServerName: Read-only name of the database server.

    databaseInstanceName        Read-only name of the database instance.

     DatabaseInstanceName: Read-only name of the database instance.

    databaseName                Read-only name of the database.

     DatabaseName: Read-only name of the database.

    failoverPartnerServerName  Read-only name of the database server.

     FailoverPartnerServerName: Read-only name of the database server.

    failoverPartnerInstanceName  Read-only name of the database server
                                instance.

     FailoverPartnerInstanceName: Read-only name of the database server
                         instance.

    multiSubnetFailover         Read-only Database MultiSubnetFailover value

      MultiSubnetFailover: Read-only Database MultiSubnetFailover value

    role                        Read-only Role of the user for this item.
                                100 is Farm Administrator, and 999 is read-
                                only. Default=999

     Role: Read-only Role of the user for this item. 100 is Farm
                         Administrator, and 999 is read-only. Default=999

  records                        If successful, records retrieved.

   PvsFarm[]: If successful, the PvsFarm object(s) are returned.
```

Mcli-Get FarmView

```
 Get the fields for a Farm View or all Farm Views in the Farm. All Farm
   Views are returned if no parameters are passed.
```

Get-PvsFarmView

```
 Get the fields for a Farm View or all Farm Views in the Farm. All Farm
   Views are returned if no parameters are passed.

  -p                               Parameters needed for the Get.

   One of these optional

    farmViewId                 GUID of the Farm View to Get.

     Guid[] Guid or FarmViewId: GUID of the Farm View to Get.

    farmViewName               Name of the Farm View to Get.

     string[] Name or FarmViewName: Name of the Farm View to Get.

    Instead of a parameter that matches one of the members listed
```

```
  -s                             Name of field to sort on. If not included,
                                 the first field listed is used. Append a - to
                                 the field name to get descending order.

   farmViewName                  name of the Farm View. Max Length=50

   description                   User description. Default="" Max Length=250

   deviceCount                   Read-only count of Devices in this Farm
                                  View. Default=0

   activeDeviceCount             Read-only count of active Devices in this
                                  Farm View. Default=0

  -f                             List of fields to retrieve, if not included,
                                 all fields are retrieved.
```

If only selected fields are needed, pass them in the Fields parameter
                     as a string array.

```
   farmViewId                    Read-only GUID that uniquely identifies this
                                  Farm View. Length=36
```

Guid or FarmViewId: Read-only GUID that uniquely identifies this Farm
                     View.

```
   farmViewName                  name of the Farm View. Max Length=50
```

Name or FarmViewName: name of the Farm View. Max Length=50

```
   description                   User description. Default="" Max Length=250
```

Description: User description. Default="" Max Length=250

```
   deviceCount                   Read-only count of Devices in this Farm
                                  View. Default=0
```

DeviceCount: Read-only count of Devices in this Farm View. Default=0

```
   activeDeviceCount             Read-only count of active Devices in this
                                  Farm View. Default=0
```

ActiveDeviceCount: Read-only count of active Devices in this Farm
                     View. Default=0

```
   makActivateNeededCount        Read-only count of active Devices that need
                                  MAK activation in this Farm View. Default=0
```

MakActivateNeededCount: Read-only count of active Devices that need
                     MAK activation in this Farm View. Default=0

```
  records                        If successful, records retrieved.
```

PvsFarmView[]: If successful, the PvsFarmView object(s) are returned.

Mcli-Get NewVersionDisks

Get new Disk versions for the Store on the Server specified.

Get-PvsNewDiskVersion

Get new Disk versions for the Store on the Server specified.

```
-p                             Parameters needed for the Get.
  One of these required
    serverId                   GUID of the Server to look for new Disk
                               versions.

      Guid[] ServerId: GUID of the Server to look for new Disk versions.

    serverName                 Name of the Server to look for new Disk
                               versions.

      string[] ServerName: Name of the Server to look for new Disk
                        versions.

  One of these required
    storeId                    GUID of the Store that the Server services
                               to look for new Disk versions.

      Guid[] StoreId: GUID of the Store that the Server services to look
                    for new Disk versions.

    storeName                  Name of the Store that the Server services
                               to look for new Disk versions.

      string[] StoreName: Name of the Store that the Server services to
                    look for new Disk versions.

  Optional
    autoAddEnabled             1 when undefined Disk versions found should
                               be automatically added, 0 otherwise.
                               Default=0

      SwitchParameter AutoAddEnabled: If -AutoAddEnabled is specified,
                    undefined Disk versions found will be automatically
                    added.

    Instead of a parameter that matches one of the members listed

      PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

        ServerId or StoreId
-s                             Name of field to sort on. If not included,
                               the first field listed is used. Append a - to
                               the field name to get descending order.

    name                       Name of the disk file without the extension.
-f                             List of fields to retrieve, if not included,
                               all fields are retrieved.

  If only selected fields are needed, pass them in the Fields parameter
                    as a string array.

    name                       Name of the disk file without the extension.

      Name: Name of the disk file without the extension.

    status                     Status of the disk file. Values are: 0
                               (Valid), 1 (Missing Properties File), 2
                               (Access Denied), 3 (Access Denied and
                               Missing Properties File), 4 (Invalid Disk
                               File), 5 (Manifest Invalid)
```

records   If successful, records retrieved.

Mcli-Get Server

Get the fields for a Server, all Servers in a Site that use a Store, service a DiskLocator, or for the whole Farm. All Servers are returned if no parameters are passed.

Get-PvsServer

Get the fields for a Server, all Servers in a Site that use a Store, service a DiskLocator, or for the whole Farm. All Servers are returned if no parameters are passed.

-p   Parameters needed for the Get.

 One of these optional

  serverId   GUID of the Server to Get.

   Guid[] Guid or ServerId: GUID of the Server to Get.

  serverName   Name of the Server to Get.

   string[] Name or ServerName: Name of the Server to Get.

  siteId   GUID of the Site to Get all Servers.

   Guid[] SiteId: GUID of the Site to Get all Servers.

  siteName   Name of the Site to Get all Servers.

   string[] SiteName: Name of the Site to Get all Servers.

  diskLocatorId   GUID of the Disk Locator to Get all Servers.

   Guid[] DiskLocatorId: GUID of the Disk Locator to Get all Servers.

 or this optional & resolution

  diskLocatorName   Name of the Disk Locator File to Get all Servers.

   string[] DiskLocatorName: Name of the Disk Locator File to Get all Servers.

 One of these optional & resolutions

  storeId   GUID of the Store to Get all Servers.

   Guid[] StoreId: GUID of the Store to Get all Servers.

  storeName   Name of the Store to Get all Servers.

   string[] StoreName: Name of the Store to Get all Servers.

 Optional

  all   Set to 1 with storeId or storeName to Get all Servers for the Store including ones with invalid paths.

One of these resolutions when needed

  siteId   GUID of the Site.

   Guid[] SiteId: GUID of the Site to Get all Servers.

  siteName   Name of the Site.

   string[] SiteName: Name of the Site to Get all Servers.

One of these resolutions when needed

  storeId   GUID of the Store that is needed when a
                    diskLocatorName is used.

   Guid[] StoreId: GUID of the Store to Get all Servers.

  storeName   Name of the Store that is needed when a
                    diskLocatorName is used.

   string[] StoreName: Name of the Store to Get all Servers.

  Instead of a parameter that matches one of the members listed

   PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

    ServerId, SiteId, DiskLocatorId or StoreId

-s   Name of field to sort on. If not included,
                    the first field listed is used. Append a - to
                    the field name to get descending order.

  serverName   Computer name with no spaces. ASCII computer
                    name characters Max Length=21

  description   User description. Default="" Max Length=250

  adMaxPasswordAge   Number of days before a password expires.
                    Min=1, Max=30, Default=7

  licenseTimeout   Amount of seconds before a license times
                    out. Min=15, Max=300, Default=30

  vDiskCreatePacing   VDisk create time pacing in miliseconds.
                    Min=0, Max=5, Default=0

  firstPort   Number of the first UDP port for use by the
                    Stream Service, First and Last must allow at
                    least 5 ports. Min=1025, Max=65534,
                    Default=6910

  lastPort   Number of the last UDP port for use by the
                    Stream Service, First and Last must allow at
                    least 5 ports. Min=1025, Max=65534,
                    Default=6930

  threadsPerPort   Number of worker threads per IO port.
                    Required that (threadPerPort * numberPorts *
                    numberIPs) <= 1000. Min=1, Max=60, Default=8

  buffersPerThread   Number of buffers per worker thread. Min=1,
                    Max=128, Default=24

serverCacheTimeout	Number of seconds to wait before considering another Server is down. Min=5, Max=60, Default=8
ioBurstSize	Number of bytes read/writes can send in a burst of packets. Required that ioBurstSize/(maxTransmissionUnits-76)<=32. Min=4096, Max=61440, Default=32768
maxTransmissionUnits	Ethernet maximum transmission unit size for the protocol for use for Server and Device. Required that ioBurstSize/(maxTransmissionUnits-76)<=32. Min=502, Max=16426, Default=1506
maxBootDevicesAllowed	Maximum number of Devices allowed to boot simultaneously. Min=1, Max=1000, Default=500
maxBootSeconds	Maximum number of seconds for a Device to boot. Min=10, Max=900, Default=60
bootPauseSeconds	Number of seconds that a Device will pause during login if its server busy. Min=1, Max=60, Default=10
adMaxPasswordAgeEnabled	Age the password, when set to 1. Default=0

| -f | List of fields to retrieve, if not included, all fields are retrieved. |

If only selected fields are needed, pass them in the Fields parameter as a string array.

| serverId | Read-only GUID that uniquely identifies this Server. Length=36 |

Guid or ServerId: Read-only GUID that uniquely identifies this Server.

| serverName | Computer name with no spaces. ASCII computer name characters Max Length=21 |

Name or ServerName: Computer name with no spaces. ASCII computer name characters Max Length=21

| siteId | GUID of the Site this Server is to be a member of. It is not used with siteName. Length=36 |

SiteId: GUID of the Site this Server is to be a member of. It is not used with SiteName.

| siteName | Name of the Site this Server is to be a member of. It is not used with siteId. |

SiteName: Name of the Site this Server is to be a member of. It is not used with SiteId.

| description | User description. Default="" Max Length=250 |

Description: User description. Default="" Max Length=250

| adMaxPasswordAge | Number of days before a password expires. Min=1, Max=30, Default=7 |

AdMaxPasswordAge: Number of days before a password expires. Min=1, Max=30, Default=7

licenseTimeout   Amount of seconds before a license times out. Min=15, Max=300, Default=30

LicenseTimeout: Amount of seconds before a license times out. Min=15, Max=300, Default=30

vDiskCreatePacing   VDisk create time pacing in miliseconds. Min=0, Max=5, Default=0

VDiskCreatePacing: VDisk create time pacing in miliseconds. Min=0, Max=5, Default=0

firstPort   Number of the first UDP port for use by the Stream Service, First and Last must allow at least 5 ports. Min=1025, Max=65534, Default=6910

FirstPort: Number of the first UDP port for use by the Stream Service, First and Last must allow at least 5 ports. Min=1025, Max=65534, Default=6910

lastPort   Number of the last UDP port for use by the Stream Service, First and Last must allow at least 5 ports. Min=1025, Max=65534, Default=6930

LastPort: Number of the last UDP port for use by the Stream Service, First and Last must allow at least 5 ports. Min=1025, Max=65534, Default=6930

threadsPerPort   Number of worker threads per IO port. Required that (threadPerPort * numberPorts * numberIPs) <= 1000. Min=1, Max=60, Default=8

ThreadsPerPort: Number of worker threads per IO port. Required that (threadPerPort * numberPorts * numberIPs) <= 1000. Min=1, Max=60, Default=8

buffersPerThread   Number of buffers per worker thread. Min=1, Max=128, Default=24

BuffersPerThread: Number of buffers per worker thread. Min=1, Max=128, Default=24

serverCacheTimeout   Number of seconds to wait before considering another Server is down. Min=5, Max=60, Default=8

ServerCacheTimeout: Number of seconds to wait before considering another Server is down. Min=5, Max=60, Default=8

ioBurstSize   Number of bytes read/writes can send in a burst of packets. Required that ioBurstSize/(maxTransmissionUnits-76)<=32. Min=4096, Max=61440, Default=32768

IoBurstSize: Number of bytes read/writes can send in a burst of packets. Required that IoBurstSize/(MaxTransmissionUnits-76)<=32. Min=4096, Max=61440, Default=32768

```
maxTransmissionUnits        Ethernet maximum transmission unit size for
                            the protocol for use for Server and Device.
                            Required that
                            ioBurstSize/(maxTransmissionUnits-76)<=32.
                            Min=502, Max=16426, Default=1506

    MaxTransmissionUnits: Ethernet maximum transmission unit size for the
                    protocol for use for Server and Device. Required
                    that IoBurstSize/(MaxTransmissionUnits-76)<=32.
                    Min=502, Max=16426, Default=1506

maxBootDevicesAllowed       Maximum number of Devices allowed to boot
                            simultaneously. Min=1, Max=1000, Default=500

    MaxBootDevicesAllowed: Maximum number of Devices allowed to boot
                    simultaneously. Min=1, Max=1000, Default=500

maxBootSeconds              Maximum number of seconds for a Device to
                            boot. Min=10, Max=900, Default=60

    MaxBootSeconds: Maximum number of seconds for a Device to boot.
                    Min=10, Max=900, Default=60

bootPauseSeconds            Number of seconds that a Device will pause
                            during login if its server busy. Min=1,
                            Max=60, Default=10

    BootPauseSeconds: Number of seconds that a Device will pause during
                    login if its server busy. Min=1, Max=60, Default=10

adMaxPasswordAgeEnabled     Age the password, when set to 1. Default=0

    AdMaxPasswordAgeEnabled: Age the password, when set to true.
                    Default=false

eventLoggingEnabled         Enable event logging, when set to 1.
                            Default=0

    EventLoggingEnabled: Enable event logging, when set to true.
                    Default=false

nonBlockingIoEnabled        Use non-Blocking IO, when set to 1.
                            Default=1

    NonBlockingIoEnabled: Use non-Blocking IO, when set to true.
                    Default=true

role                        Read-only Role of the user for this item.
                            100 is Farm Administrator, and 200 is Site
                            Administrator. Default=999

    Role: Read-only Role of the user for this item. 100 is Farm
                    Administrator, and 200 is Site Administrator.
                    Default=999

ip                          One or more streaming ip addresses. If more
                            than one ip is included, the ip addresses
                            are comma delimited.

    Ip: One or more streaming ip addresses.

initialQueryConnectionPoolSize  Initial size of database connection
                            pool for non-transactional queries. Min=1,
                            Max=1000, Default=50
```

InitialQueryConnectionPoolSize: Initial size of database connection pool for non-transactional queries. Min=1, Max=1000, Default=50

initialTransactionConnectionPoolSize Initial size of database connection pool for transactional queries. Min=1, Max=1000, Default=50

InitialTransactionConnectionPoolSize: Initial size of database connection pool for transactional queries. Min=1, Max=1000, Default=50

maxQueryConnectionPoolSize Maximum size of database connection pool for non-transactional queries. Min=1, Max=32767, Default=1000

MaxQueryConnectionPoolSize: Maximum size of database connection pool for non-transactional queries. Min=1, Max=32767, Default=1000

maxTransactionConnectionPoolSize Maximum size of database connection pool for transactional queries. Min=1, Max=32767, Default=1000

MaxTransactionConnectionPoolSize: Maximum size of database connection pool for transactional queries. Min=1, Max=32767, Default=1000

refreshInterval Interval, in number of seconds, the server should wait before refreshing settings. If set to 0, unused database connections are never released. Min=0, Max=32767, Default=300

RefreshInterval: Interval, in number of seconds, the server should wait before refreshing settings. If set to 0, unused database connections are never released. Min=0, Max=32767, Default=300

unusedDbConnectionTimeout Interval, in number of seconds, a connection should go unused before it is to be released. Min=0, Max=32767, Default=300

UnusedDbConnectionTimeout: Interval, in number of seconds, a connection should go unused before it is to be released. Min=0, Max=32767, Default=300

busyDbConnectionRetryCount Number of times a failed database connection will be retried. Min=0, Max=32767, Default=2

BusyDbConnectionRetryCount: Number of times a failed database connection will be retried. Min=0, Max=32767, Default=2

busyDbConnectionRetryInterval Interval, in number of milliseconds, the server should wait before retrying to connect to a database. Min=0, Max=10000, Default=25

BusyDbConnectionRetryInterval: Interval, in number of milliseconds, the server should wait before retrying to connect to a database. Min=0, Max=10000, Default=25

localConcurrentIoLimit	Maximum concurrent IO transactions it performs for vDisks that are local. A value of 0 disables the feature. Min=0, Max=128, Default=4

LocalConcurrentIoLimit: Maximum concurrent IO transactions it performs for vDisks that are local. A value of 0 disables the feature. Min=0, Max=128, Default=4

remoteConcurrentIoLimit	Maximum concurrent IO transactions it performs for vDisks that are remote. A value of 0 disables the feature. Min=0, Max=128, Default=4

RemoteConcurrentIoLimit: Maximum concurrent IO transactions it performs for vDisks that are remote. A value of 0 disables the feature. Min=0, Max=128, Default=4

active	1 if the Server is currently active, 2 if unknown, and 0 otherwise. Min=0, Max=2, Default=0

Active: 1 if the Server is currently active, 2 if unknown, and 0 otherwise. Min=0, Max=2, Default=0

logLevel	Level to perform logging at. Values are: 0 (None), 1 (Fatal), 2 (Error), 3 (Warning), 4 (Info), 5 (Debug), and 6 (Trace). Min=0, Max=6, Default=4

LogLevel: Level to perform logging at. Values are: 0 (None), 1 (Fatal), 2 (Error), 3 (Warning), 4 (Info), 5 (Debug), and 6 (Trace). Min=0, Max=6, Default=4

logFileSizeMax	Maximum size log files can reach in Megabytes. Min=1, Max=50, Default=5

LogFileSizeMax: Maximum size log files can reach in Megabytes. Min=1, Max=50, Default=5

logFileBackupCopiesMax	Maximum number of log file backups. Min=1, Max=50, Default=4

LogFileBackupCopiesMax: Maximum number of log file backups. Min=1, Max=50, Default=4

powerRating	A strictly relative rating of this Server's capabilities when compared to other Servers in the Store(s) it belongs too; can be used to help tune load balancing. Min=0.1, Max=1000, Default=1

PowerRating: A strictly relative rating of this Server's capabilities when compared to other Servers in the Store(s) it belongs too; can be used to help tune load balancing. Min=0.1, Max=1000, Default=1

serverFqdn	Read-only fully qualified domain name. Default=null Max Length=1024

ServerFqdn: Read-only fully qualified domain name. Default="" Max Length=1024

```
        managementIp              IP address used for management
                                  communications between Servers.
                                  Default=0.0.0.0

     ManagementIp: IP address used for management communications between
                    Servers. Default=0.0.0.0

        lastCeipUploadAttempt     Time that this server last attempted a CEIP
                                  upload. Default=null

     LastCeipUploadAttempt: Time that this server last attempted a CEIP
                    upload. Default=Empty

        lastBugReportAttempt      Time that this server last attempted to
                                  upload or generate a bug report bundle.
                                  Default=null

     LastBugReportAttempt: Time that this server last attempted to upload
                    or generate a bug report bundle. Default=Empty

        lastBugReportStatus       Status of the last bug report on this
                                  server. Default=null Max Length=250

     LastBugReportStatus: Status of the last bug report on this server.
                    Default="" Max Length=250

        lastBugReportResult       Status of the last bug report on this
                                  server. Default=null Max Length=4000

     LastBugReportResult: Status of the last bug report on this server.
                    Default="" Max Length=4000

        lastBugReportSummary      Summary of the last bug report on this
                                  server. Default=null Max Length=250

     LastBugReportSummary: Summary of the last bug report on this server.
                    Default="" Max Length=250

     records                      If successful, records retrieved.

      PvsServer[]: If successful, the PvsServer object(s) are returned.
```

Mcli-Get ServerBiosBootstrap

```
    Oem Only: Get the bootstrap fields for the Server dell_bios.bin BIOS
       bootstrap file.
```

Get-PvsServerBiosBootstrap

```
    Oem Only: Get the bootstrap fields for the Server dell_bios.bin BIOS
       bootstrap file.

     -p                           Parameters needed for the Get.

      One of these required

        serverId                  GUID of the Server to Get the dell_bios.bin
                                  BIOS bootstrap file from.

         Guid[] Guid or ServerId: GUID of the Server to Get the dell_bios.bin
                    BIOS bootstrap file from.

        serverName                Name of the Server to Get the dell_bios.bin
                                  BIOS bootstrap file from.
```

string[] Name or ServerName: Name of the Server to Get the
dell_bios.bin BIOS bootstrap file from.

Instead of a parameter that matches one of the members listed

PvsObject[] Object: PvsObjects with the members below can be used as
the Object parameter or from a pileline:

ServerId

-f   List of fields to retrieve, if not included,
                               all fields are retrieved.

If only selected fields are needed, pass them in the Fields parameter
as a string array.

enabled   Automatically update the BIOS on the target
                               device with these setting when set to 1,
                               otherwise do not use these settings.
                               Default=0

Enabled: Automatically update the BIOS on the target device with
these setting when set to true, otherwise do not use
these settings. Default=false

dhcpEnabled   Use DHCP to retrieve target device IP when
                               set to 1, otherwise use the static domain,
                               dnsIpAddress1 and dnsIpAddress2 settings.
                               Default=1

DhcpEnabled: Use DHCP to retrieve target device IP when set to true,
otherwise use the static domain, dnsIpAddresstrue
and dnsIpAddress2 settings. Default=true

lookup   Use DNS to find the Server when set to 1
                               with the serverName host value, otherwise
                               use the bootserver1_Ip, bootserver1_Port,
                               bootserver2_Ip, bootserver2_Port,
                               bootserver3_Ip, bootserver3_Port,
                               bootserver4_Ip, and bootserver4_Port
                               settings. Default=1

Lookup: Use DNS to find the Server when set to true with the
ServerName host value, otherwise use the
bootservertrue_Ip, bootservertrue_Port,
bootserver2_Ip, bootserver2_Port, bootserver3_Ip,
bootserver3_Port, bootserver4_Ip, and
bootserver4_Port settings. Default=true

verboseMode   Display verbose diagnostic information when
                               set to 1. Default=0

VerboseMode: Display verbose diagnostic information when set to true.
Default=false

interruptSafeMode   Interrupt safe mode (use if target device
                               hangs during boot) when set to 1. Default=0

InterruptSafeMode: Interrupt safe mode (use if target device hangs
during boot) when set to true. Default=false

paeMode   PAE mode (use if PAE enabled in boot.ini of
                               target device) when set to 1. Default=0

PaeMode: PAE mode (use if PAE enabled in boot.ini of target device) when set to true. Default=false

bootFromHdOnFail   For network recovery reboot to hard drive when set to 1, restore network connection when set to 0. Default=0

BootFromHdOnFail: For network recovery reboot to hard drive when set to true, restore network connection when set to false. Default=false

recoveryTime   When bootFromHdOnFail is 1, this is the number of seconds to wait before reboot to hard drive. Min=10, Max=60000, Default=50

RecoveryTime: When bootFromHdOnFail is 1, this is the number of seconds to wait before reboot to hard drive. Min=10, Max=60000, Default=50

pollingTimeout   Login polling timeout in milliseconds. Min=1000, Max=60000, Default=5000

PollingTimeout: Login polling timeout in milliseconds. Min=1000, Max=60000, Default=5000

generalTimeout   Login general timeout in milliseconds. Min=1000, Max=60000, Default=5000

GeneralTimeout: Login general timeout in milliseconds. Min=1000, Max=60000, Default=5000

serverName   Host to use for DNS lookup. Only used when lookup is 1. Default=IMAGESERVER1

Name or ServerName: Host to use for DNS lookup. Only used when Lookup is true. Default=IMAGESERVER1

bootserver1_Ip   1st boot server IP. Only used when lookup is 0.

Bootserver1_Ip: 1st boot server IP. Only used when Lookup is false.

bootserver1_Port   1st boot server port. Only used when lookup is 0. Min=1025, Max=65536, Default=6910

Bootserver1_Port: 1st boot server port. Only used when Lookup is false. Min=1025, Max=65536, Default=6910

bootserver2_Ip   2nd boot server IP. Only used when lookup is 0. Default=0.0.0.0

Bootserver2_Ip: 2nd boot server IP. Only used when Lookup is false. Default=0.0.0.0

bootserver2_Port   2nd boot server port. Only used when lookup is 0. Min=1025, Max=65536, Default=6910

Bootserver2_Port: 2nd boot server port. Only used when Lookup is false. Min=1025, Max=65536, Default=6910

bootserver3_Ip   3rd boot server IP. Only used when lookup is 0. Default=0.0.0.0

Bootserver3_Ip: 3rd boot server IP. Only used when Lookup is false. Default=0.0.0.0

```
        bootserver3_Port              3rd boot server port. Only used when lookup
                                      is 0. Min=1025, Max=65536, Default=6910

        Bootserver3_Port: 3rd boot server port. Only used when Lookup is
                          false. Min=1025, Max=65536, Default=6910

        bootserver4_Ip                4th boot server IP. Only used when lookup is
                                      0. Default=0.0.0.0

        Bootserver4_Ip: 4th boot server IP. Only used when Lookup is false.
                        Default=0.0.0.0

        bootserver4_Port              4th boot server port. Only used when lookup
                                      is 0. Min=1025, Max=65536, Default=6910

        Bootserver4_Port: 4th boot server port. Only used when Lookup is
                          false. Min=1025, Max=65536, Default=6910

        domain                        Domain of the primary and secondary DNS
                                      servers. Only used when dhcpEnabled is 0.

        Domain: Domain of the primary and secondary DNS servers. Only used
                when DhcpEnabled is false.

        dnsIpAddress1                 Primary DNS server IP. Only used when
                                      dhcpEnabled is 0.

        DnsIpAddress1: Primary DNS server IP. Only used when DhcpEnabled is
                       false.

        dnsIpAddress2                 Secondary DNS server IP. Only used when
                                      dhcpEnabled is 0.

        DnsIpAddress2: Secondary DNS server IP. Only used when DhcpEnabled is
                       false.

      records                         If successful, records retrieved.

        PvsServerBiosBootstrap[]: If successful, the PvsServerBiosBootstrap
                         object(s) are returned.
```

Mcli-Get ServerBootstrap

Get the bootstrap fields for the Server and named bootstrap file specified.

Get-PvsServerBootstrap

Get the bootstrap fields for the Server and named bootstrap file specified.

```
     -p                              Parameters needed for the Get.

      One of these required

        serverId                      GUID of the Server to Get the named
                                      bootstrap file from.

         Guid[] Guid or ServerId: GUID of the Server to Get the named
                        bootstrap file from.

        serverName                    Name of the Server to Get the named
                                      bootstrap file from.

         string[] ServerName: Name of the Server to Get the named bootstrap
                        file from.

      This required
```

```
name                        Name of the bootstrap file.

   string[] Name: Name of the bootstrap file.

   Instead of a parameter that matches one of the members listed

   PvsObject[] Object: PvsObjects with the members below can be used as
                the Object parameter or from a pileline:

     (ServerId and Name) or ServerId

 -f                           List of fields to retrieve, if not included,
                              all fields are retrieved.

  If only selected fields are needed, pass them in the Fields parameter
                  as a string array.

   bootserver1_Ip           1st boot server IP.

    Bootserver1_Ip: 1st boot server IP.

   bootserver1_Netmask      1st boot server netmask. Default=0.0.0.0

    Bootserver1_Netmask: 1st boot server netmask. Default=0.0.0.0

   bootserver1_Gateway      1st boot server gateway. Default=0.0.0.0

    Bootserver1_Gateway: 1st boot server gateway. Default=0.0.0.0

   bootserver1_Port         1st boot server port. Min=1025, Max=65536,
                            Default=6910

    Bootserver1_Port: 1st boot server port. Min=1025, Max=65536,
                Default=6910

   bootserver2_Ip           2nd boot server IP. Default=0.0.0.0

    Bootserver2_Ip: 2nd boot server IP. Default=0.0.0.0

   bootserver2_Netmask      2nd boot server netmask. Default=0.0.0.0

    Bootserver2_Netmask: 2nd boot server netmask. Default=0.0.0.0

   bootserver2_Gateway      2nd boot server gateway. Default=0.0.0.0

    Bootserver2_Gateway: 2nd boot server gateway. Default=0.0.0.0

   bootserver2_Port         2nd boot server port. Min=1025, Max=65536,
                            Default=6910

    Bootserver2_Port: 2nd boot server port. Min=1025, Max=65536,
                Default=6910

   bootserver3_Ip           3rd boot server IP. Default=0.0.0.0

    Bootserver3_Ip: 3rd boot server IP. Default=0.0.0.0

   bootserver3_Netmask      3rd boot server netmask. Default=0.0.0.0

    Bootserver3_Netmask: 3rd boot server netmask. Default=0.0.0.0

   bootserver3_Gateway      3rd boot server gateway. Default=0.0.0.0

    Bootserver3_Gateway: 3rd boot server gateway. Default=0.0.0.0

   bootserver3_Port         3rd boot server port. Min=1025, Max=65536,
                            Default=6910

    Bootserver3_Port: 3rd boot server port. Min=1025, Max=65536,
                Default=6910
```

bootserver4_Ip   4th boot server IP. Default=0.0.0.0

Bootserver4_Ip: 4th boot server IP. Default=0.0.0.0

bootserver4_Netmask   4th boot server netmask. Default=0.0.0.0

Bootserver4_Netmask: 4th boot server netmask. Default=0.0.0.0

bootserver4_Gateway   4th boot server gateway. Default=0.0.0.0

Bootserver4_Gateway: 4th boot server gateway. Default=0.0.0.0

bootserver4_Port   4th boot server port. Min=1025, Max=65536,
                                 Default=6910

Bootserver4_Port: 4th boot server port. Min=1025, Max=65536,
                 Default=6910

verboseMode   Display verbose diagnostic information when
                                 set to 1. Default=0

VerboseMode: Display verbose diagnostic information when set to true.
                 Default=false

interruptSafeMode   Interrupt safe mode (use if target device
                                 hangs during boot) when set to 1. Default=0

InterruptSafeMode: Interrupt safe mode (use if target device hangs
                 during boot) when set to true. Default=false

paeMode   PAE mode (use if PAE enabled in boot.ini of
                                 target device) when set to 1. Default=0

PaeMode: PAE mode (use if PAE enabled in boot.ini of target device)
                 when set to true. Default=false

bootFromHdOnFail   For network recovery reboot to hard drive
                                 when set to 1, restore network connection
                                 when set to 0. Default=0

BootFromHdOnFail: For network recovery reboot to hard drive when set
                 to true, restore network connection when set to
                 false. Default=false

recoveryTime   When bootFromHdOnFail is 1, this is the
                                 number of seconds to wait before reboot to
                                 hard drive. Min=10, Max=60000, Default=50

RecoveryTime: When bootFromHdOnFail is 1, this is the number of
                 seconds to wait before reboot to hard drive. Min=10,
                 Max=60000, Default=50

pollingTimeout   Login polling timeout in milliseconds.
                                 Min=1000, Max=60000, Default=5000

PollingTimeout: Login polling timeout in milliseconds. Min=1000,
                 Max=60000, Default=5000

generalTimeout   Login general timeout in milliseconds.
                                 Min=1000, Max=60000, Default=5000

GeneralTimeout: Login general timeout in milliseconds. Min=1000,
                 Max=60000, Default=5000

records   If successful, records retrieved.

PvsServerBootstrap[]: If successful, the PvsServerBootstrap object(s)
                 are returned.

Mcli-Get ServerBootstrapNames

Get the bootstrap names for a Server.

Get-PvsServerBootstrapName

Get the bootstrap names for a Server.

   -p   Parameters needed for the Get.

  One of these required

   serverId   GUID of the Server to Get bootstrap names for.

    Guid[] Guid or ServerId: GUID of the Server to Get bootstrap names for.

   serverName   Name of the Server to Get bootstrap names for.

    string[] Name or ServerName: Name of the Server to Get bootstrap names for.

  Instead of a parameter that matches one of the members listed

    PvsObject[] Object: PvsObjects with the members below can be used as the Object parameter or from a pileline:

     ServerId

  -f   List of fields to retrieve, if not included, all fields are retrieved.

   name   Bootstrap file name.

    Name: Bootstrap file name.

  records   If successful, records retrieved.

    PvsServerBootstrapName[]: If successful, the PvsServerBootstrapName object(s) are returned.

Mcli-Get ServerInfo

Get the fields and status for a Server, all Servers in a Site that use a Store, service a DiskLocator, or for the whole Farm. All Servers are returned if no parameters are passed.

Get-PvsServerInfo

Get the fields and status for a Server, all Servers in a Site that use a Store, service a DiskLocator, or for the whole Farm. All Servers are returned if no parameters are passed.

   -p   Parameters needed for the Get.

  One of these optional

   serverId   GUID of the Server to Get.

    Guid[] Guid or ServerId: GUID of the Server to Get.

   serverName   Name of the Server to Get.

    string[] Name or ServerName: Name of the Server to Get.

   siteId   GUID of the Site to Get all Servers.

      Guid[] SiteId: GUID of the Site to Get all Servers.

  siteName   Name of the Site to Get all Servers.

    string[] SiteName: Name of the Site to Get all Servers.

  diskLocatorId   GUID of the Disk Locator to Get all Servers.

    Guid[] DiskLocatorId: GUID of the Disk Locator to Get all Servers.

or this optional & resolution

  diskLocatorName   Name of the Disk Locator File to Get all
                             Servers.

    string[] DiskLocatorName: Name of the Disk Locator File to Get all
                          Servers.

One of these optional & resolutions

  storeId   GUID of the Store to Get all Servers.

    Guid[] StoreId: GUID of the Store to Get all Servers.

  storeName   Name of the Store to Get all Servers.

    string[] StoreName: Name of the Store to Get all Servers.

Optional

  all   Set to 1 with storeId or storeName to Get
                          all Servers for the Store including ones
                          with invalid paths.

    SwitchParameter All: If -All is specified, with StoreId or StoreName
                  all Servers for the Store including ones with
                  invalid paths will be returned.

One of these resolutions when needed

  siteId   GUID of the Site.

    Guid[] SiteId: GUID of the Site to Get all Servers.

  siteName   Name of the Site.

    string[] SiteName: Name of the Site to Get all Servers.

One of these resolutions when needed

  storeId   GUID of the Store that is needed when a
                        diskLocatorName is used.

    Guid[] StoreId: GUID of the Store to Get all Servers.

  storeName   Name of the Store that is needed when a
                        diskLocatorName is used.

    string[] StoreName: Name of the Store to Get all Servers.

  Instead of a parameter that matches one of the members listed

    PvsObject[] Object: PvsObjects with the members below can be used as
                  the Object parameter or from a pileline:

      ServerId, SiteId, DiskLocatorId or StoreId

-s   Name of field to sort on. If not included,
                        the first field listed is used. Append a - to
                        the field name to get descending order.

serverName	Computer name with no spaces. ASCII computer name characters Max Length=21
description	User description. Default="" Max Length=250
adMaxPasswordAge	Number of days before a password expires. Min=1, Max=30, Default=7
licenseTimeout	Amount of seconds before a license times out. Min=15, Max=300, Default=30
vDiskCreatePacing	VDisk create time pacing in miliseconds. Min=0, Max=5, Default=0
firstPort	Number of the first UDP port for use by the Stream Service, First and Last must allow at least 5 ports. Min=1025, Max=65534, Default=6910
lastPort	Number of the last UDP port for use by the Stream Service, First and Last must allow at least 5 ports. Min=1025, Max=65534, Default=6930
threadsPerPort	Number of worker threads per IO port. Required that (threadPerPort * numberPorts * numberIPs) <= 1000. Min=1, Max=60, Default=8
buffersPerThread	Number of buffers per worker thread. Min=1, Max=128, Default=24
serverCacheTimeout	Number of seconds to wait before considering another Server is down. Min=5, Max=60, Default=8
ioBurstSize	Number of bytes read/writes can send in a burst of packets. Required that ioBurstSize/(maxTransmissionUnits-76)<=32. Min=4096, Max=61440, Default=32768
maxTransmissionUnits	Ethernet maximum transmission unit size for the protocol for use for Server and Device. Required that ioBurstSize/(maxTransmissionUnits-76)<=32. Min=502, Max=16426, Default=1506
maxBootDevicesAllowed	Maximum number of Devices allowed to boot simultaneously. Min=1, Max=1000, Default=500
maxBootSeconds	Maximum number of seconds for a Device to boot. Min=10, Max=900, Default=60
bootPauseSeconds	Number of seconds that a Device will pause during login if its server busy. Min=1, Max=60, Default=10
adMaxPasswordAgeEnabled	Age the password, when set to 1. Default=0
-f	List of fields to retrieve, if not included, all fields are retrieved.

If only selected fields are needed, pass them in the Fields parameter as a string array.

serverId   Read-only GUID that uniquely identifies this
                            Server. Length=36

 Guid or ServerId: Read-only GUID that uniquely identifies this
                Server.

serverName   Computer name with no spaces. ASCII computer
                            name characters Max Length=21

 Name or ServerName: Computer name with no spaces. ASCII computer name
                characters Max Length=21

siteId   GUID of the Site this Server is to be a
                            member of. It is not used with siteName.
                            Length=36

 SiteId: GUID of the Site this Server is to be a member of. It is not
                used with SiteName.

siteName   Name of the Site this Server is to be a
                            member of. It is not used with siteId.

 SiteName: Name of the Site this Server is to be a member of. It is
                not used with SiteId.

description   User description. Default="" Max Length=250

 Description: User description. Default="" Max Length=250

adMaxPasswordAge   Number of days before a password expires.
                            Min=1, Max=30, Default=7

 AdMaxPasswordAge: Number of days before a password expires. Min=1,
                Max=30, Default=7

licenseTimeout   Amount of seconds before a license times
                            out. Min=15, Max=300, Default=30

 LicenseTimeout: Amount of seconds before a license times out. Min=15,
                Max=300, Default=30

vDiskCreatePacing   VDisk create time pacing in miliseconds.
                            Min=0, Max=5, Default=0

 VDiskCreatePacing: VDisk create time pacing in miliseconds. Min=0,
                Max=5, Default=0

firstPort   Number of the first UDP port for use by the
                            Stream Service, First and Last must allow at
                            least 5 ports. Min=1025, Max=65534,
                            Default=6910

 FirstPort: Number of the first UDP port for use by the Stream
                Service, First and Last must allow at least 5 ports.
                Min=1025, Max=65534, Default=6910

lastPort   Number of the last UDP port for use by the
                            Stream Service, First and Last must allow at
                            least 5 ports. Min=1025, Max=65534,
                            Default=6930

 LastPort: Number of the last UDP port for use by the Stream Service,
                First and Last must allow at least 5 ports.
                Min=1025, Max=65534, Default=6930

threadsPerPort   Number of worker threads per IO port.
                                Required that (threadPerPort * numberPorts *
                                numberIPs) <= 1000. Min=1, Max=60, Default=8

   ThreadsPerPort: Number of worker threads per IO port. Required that
                   (threadPerPort * numberPorts * numberIPs) <= 1000.
                   Min=1, Max=60, Default=8

buffersPerThread   Number of buffers per worker thread. Min=1,
                                Max=128, Default=24

   BuffersPerThread: Number of buffers per worker thread. Min=1,
                     Max=128, Default=24

serverCacheTimeout   Number of seconds to wait before considering
                                another Server is down. Min=5, Max=60,
                                Default=8

   ServerCacheTimeout: Number of seconds to wait before considering
                       another Server is down. Min=5, Max=60, Default=8

ioBurstSize   Number of bytes read/writes can send in a
                                burst of packets. Required that
                                ioBurstSize/(maxTransmissionUnits-76)<=32.
                                Min=4096, Max=61440, Default=32768

   IoBurstSize: Number of bytes read/writes can send in a burst of
                packets. Required that
                IoBurstSize/(MaxTransmissionUnits-76)<=32. Min=4096,
                Max=61440, Default=32768

maxTransmissionUnits   Ethernet maximum transmission unit size for
                                the protocol for use for Server and Device.
                                Required that
                                ioBurstSize/(maxTransmissionUnits-76)<=32.
                                Min=502, Max=16426, Default=1506

   MaxTransmissionUnits: Ethernet maximum transmission unit size for the
                         protocol for use for Server and Device. Required
                         that IoBurstSize/(MaxTransmissionUnits-76)<=32.
                         Min=502, Max=16426, Default=1506

maxBootDevicesAllowed   Maximum number of Devices allowed to boot
                                simultaneously. Min=1, Max=1000, Default=500

   MaxBootDevicesAllowed: Maximum number of Devices allowed to boot
                          simultaneously. Min=1, Max=1000, Default=500

maxBootSeconds   Maximum number of seconds for a Device to
                                boot. Min=10, Max=900, Default=60

   MaxBootSeconds: Maximum number of seconds for a Device to boot.
                   Min=10, Max=900, Default=60

bootPauseSeconds   Number of seconds that a Device will pause
                                during login if its server busy. Min=1,
                                Max=60, Default=10

   BootPauseSeconds: Number of seconds that a Device will pause during
                     login if its server busy. Min=1, Max=60, Default=10

adMaxPasswordAgeEnabled   Age the password, when set to 1. Default=0

   AdMaxPasswordAgeEnabled: Age the password, when set to true.
                            Default=false

```
eventLoggingEnabled        Enable event logging, when set to 1.
                           Default=0
```

EventLoggingEnabled: Enable event logging, when set to true.
Default=false

```
nonBlockingIoEnabled       Use non-Blocking IO, when set to 1.
                           Default=1
```

NonBlockingIoEnabled: Use non-Blocking IO, when set to true.
Default=true

```
role                       Read-only Role of the user for this item.
                           100 is Farm Administrator, and 200 is Site
                           Administrator. Default=999
```

Role: Read-only Role of the user for this item. 100 is Farm
Administrator, and 200 is Site Administrator.
Default=999

```
ip                         One or more streaming ip addresses. If more
                           than one ip is included, the ip addresses
                           are comma delimited.
```

Ip: One or more streaming ip addresses.

```
initialQueryConnectionPoolSize  Initial size of database connection
                           pool for non-transactional queries. Min=1,
                           Max=1000, Default=50
```

InitialQueryConnectionPoolSize: Initial size of database connection
pool for non-transactional queries. Min=1, Max=1000,
Default=50

```
initialTransactionConnectionPoolSize  Initial size of database
                           connection pool for transactional queries.
                           Min=1, Max=1000, Default=50
```

InitialTransactionConnectionPoolSize: Initial size of database
connection pool for transactional queries. Min=1,
Max=1000, Default=50

```
maxQueryConnectionPoolSize  Maximum size of database connection pool
                           for non-transactional queries. Min=1,
                           Max=32767, Default=1000
```

MaxQueryConnectionPoolSize: Maximum size of database connection pool
for non-transactional queries. Min=1, Max=32767,
Default=1000

```
maxTransactionConnectionPoolSize  Maximum size of database connection
                           pool for transactional queries. Min=1,
                           Max=32767, Default=1000
```

MaxTransactionConnectionPoolSize: Maximum size of database connection
pool for transactional queries. Min=1, Max=32767,
Default=1000

```
refreshInterval            Interval, in number of seconds, the server
                           should wait before refreshing settings. If
                           set to 0, unused database connections are
                           never released. Min=0, Max=32767,
                           Default=300
```

RefreshInterval: Interval, in number of seconds, the server should wait before refreshing settings. If set to 0, unused database connections are never released. Min=0, Max=32767, Default=300

unusedDbConnectionTimeout Interval, in number of seconds, a connection should go unused before it is to be released. Min=0, Max=32767, Default=300

UnusedDbConnectionTimeout: Interval, in number of seconds, a connection should go unused before it is to be released. Min=0, Max=32767, Default=300

busyDbConnectionRetryCount Number of times a failed database connection will be retried. Min=0, Max=32767, Default=2

BusyDbConnectionRetryCount: Number of times a failed database connection will be retried. Min=0, Max=32767, Default=2

busyDbConnectionRetryInterval Interval, in number of milliseconds, the server should wait before retrying to connect to a database. Min=0, Max=10000, Default=25

BusyDbConnectionRetryInterval: Interval, in number of milliseconds, the server should wait before retrying to connect to a database. Min=0, Max=10000, Default=25

localConcurrentIoLimit Maximum concurrent IO transactions it performs for vDisks that are local. A value of 0 disables the feature. Min=0, Max=128, Default=4

LocalConcurrentIoLimit: Maximum concurrent IO transactions it performs for vDisks that are local. A value of 0 disables the feature. Min=0, Max=128, Default=4

remoteConcurrentIoLimit Maximum concurrent IO transactions it performs for vDisks that are remote. A value of 0 disables the feature. Min=0, Max=128, Default=4

RemoteConcurrentIoLimit: Maximum concurrent IO transactions it performs for vDisks that are remote. A value of 0 disables the feature. Min=0, Max=128, Default=4

active 1 if the Server is currently active, 2 if unknown, and 0 otherwise. Min=0, Max=2, Default=0

Active: 1 if the Server is currently active, 2 if unknown, and 0 otherwise. Min=0, Max=2, Default=0

logLevel Level to perform logging at. Values are: 0 (None), 1 (Fatal), 2 (Error), 3 (Warning), 4 (Info), 5 (Debug), and 6 (Trace). Min=0, Max=6, Default=4

LogLevel: Level to perform logging at. Values are: 0 (None), 1 (Fatal), 2 (Error), 3 (Warning), 4 (Info), 5 (Debug), and 6 (Trace). Min=0, Max=6, Default=4

```
logFileSizeMax              Maximum size log files can reach in
                            Megabytes. Min=1, Max=50, Default=5
```

LogFileSizeMax: Maximum size log files can reach in Megabytes. Min=1,
                Max=50, Default=5

```
logFileBackupCopiesMax      Maximum number of log file backups. Min=1,
                            Max=50, Default=4
```

LogFileBackupCopiesMax: Maximum number of log file backups. Min=1,
                Max=50, Default=4

```
powerRating                 A strictly relative rating of this Server's
                            capabilities when compared to other Servers
                            in the Store(s) it belongs too; can be used
                            to help tune load balancing. Min=0.1,
                            Max=1000, Default=1
```

PowerRating: A strictly relative rating of this Server's capabilities
                when compared to other Servers in the Store(s) it
                belongs too; can be used to help tune load
                balancing. Min=0.1, Max=1000, Default=1

```
serverFqdn                  Read-only fully qualified domain name.
                            Default=null Max Length=1024
```

ServerFqdn: Read-only fully qualified domain name. Default="" Max
                Length=1024

```
managementIp                IP address used for management
                            communications between Servers.
                            Default=0.0.0.0
```

ManagementIp: IP address used for management communications between
                Servers. Default=0.0.0.0

```
lastCeipUploadAttempt       Time that this server last attempted a CEIP
                            upload. Default=null
```

LastCeipUploadAttempt: Time that this server last attempted a CEIP
                upload. Default=Empty

```
lastBugReportAttempt        Time that this server last attempted to
                            upload or generate a bug report bundle.
                            Default=null
```

LastBugReportAttempt: Time that this server last attempted to upload
                or generate a bug report bundle. Default=Empty

```
lastBugReportStatus         Status of the last bug report on this
                            server. Default=null Max Length=250
```

LastBugReportStatus: Status of the last bug report on this server.
                Default="" Max Length=250

```
lastBugReportResult         Status of the last bug report on this
                            server. Default=null Max Length=4000
```

LastBugReportResult: Status of the last bug report on this server.
                Default="" Max Length=4000

```
lastBugReportSummary        Summary of the last bug report on this
                            server. Default=null Max Length=250
```

LastBugReportSummary: Summary of the last bug report on this server.
                Default="" Max Length=250

```
   contactIp                   Read-only contact IP for the Server.

    ContactIp: Read-only contact IP for the Server.
   contactPort                 Read-only contact port for the Server.

    ContactPort: Read-only contact port for the Server.
   deviceCount                 Read-only count of Devices. Default=0

    DeviceCount: Read-only count of Devices. Default=0
 records                       If successful, records retrieved.

  PvsServerInfo[]: If successful, the PvsServerInfo object(s) are
                   returned.
```

Mcli-Get ServerStatus

```
  Get the Server Status fields for a Server.
```

Get-PvsServerStatus

```
  Get the Server Status fields for a Server.
   -p                          Parameters needed for the Get.
    One of these required
    serverId                   GUID of the Server to Get status for.

     Guid[] Guid or ServerId: GUID of the Server to Get status for.
    serverName                 Name of Server to Get status for.

     string[] Name or ServerName: Name of Server to Get status for.
    Instead of a parameter that matches one of the members listed

     PvsObject[] Object: PvsObjects with the members below can be used as
                   the Object parameter or from a pileline:

       ServerId
   -f                          List of fields to retrieve, if not included,
                               all fields are retrieved.

   If only selected fields are needed, pass them in the Fields parameter
                   as a string array.
    serverId                   Read-only GUID of the Server. Can be used
                                with Get Server. Length=36

     Guid or ServerId: Read-only GUID of the Server. Can be used with Get
                   Server.
    serverName                 Read-only Name of the Server. Can be used
                                with Get Server.

     Name or ServerName: Read-only Name of the Server. Can be used with
                   Get Server.
    ip                         Read-only contact IP for the Server.

     Ip: Read-only contact IP for the Server.
    port                       Read-only contact port for the Server.

     Port: Read-only contact port for the Server.
```

```
    deviceCount                Read-only count of Devices. Default=0

        DeviceCount: Read-only count of Devices. Default=0

    status                     Status of the server, 0 if down, 1 if up and
                               2 if unknown. Default=0

        Status: Status of the server, 0 if down, 1 if up and 2 if unknown.
                    Default=0

  records                      If successful, records retrieved.

      PvsServerStatus[]: If successful, the PvsServerStatus object(s) are
                    returned.
```

Mcli-Get ServerStore

Get the directory and cache paths of a Server for one or all Stores.

Get-PvsServerStore

```
    Get the directory and cache paths of a Server for one or all Stores.
    -p                         Parameters needed for the Get.
     One of these required
     serverId                  GUID of a Server.

        Guid[] ServerId: GUID of a Server.

     serverName                Name of a Server.

        string[] ServerName: Name of a Server.

     One of these optional
     storeId                   GUID of the Store.

        Guid[] StoreId: GUID of the Store.

     storeName                 Name of the Store.

        string[] StoreName: Name of the Store.

     Instead of a parameter that matches one of the members listed

        PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

          (ServerId and StoreId), ServerId or StoreId
    -f                         List of fields to retrieve, if not included,
                               all fields are retrieved.

      If only selected fields are needed, pass them in the Fields parameter
                    as a string array.
     storeId                   GUID of the Store. storeName can be used
                               instead. Length=36

        StoreId: GUID of the Store. StoreName can be used instead.

     storeName                 Name of the Store. storeId can be used
                               instead.

        StoreName: Name of the Store. StoreId can be used instead.

     serverId                  GUID of the server that uses the Store.
                               serverName can be used instead. Length=36
```

serverName   Name of the server that uses the Store.
                                serverId can be used instead.

path   Directory path that the Server uses to
                                access the Store. Default="" Max Length=255

cachePath   Cache path(s) that the Server uses with the
                                Store. It is an ordered comma-delimited
                                list. If none are specified the caches will
                                be placed in the Store cachePath.
                                Default=None

records   If successful, records retrieved.

PvsServerStore[]: If successful, the PvsServerStore object(s) are returned.

Mcli-Get Site

Get the fields for a Site or all Sites. All Sites are returned if no
parameters are passed.

Get-PvsSite

Get the fields for a Site or all Sites. All Sites are returned if no
parameters are passed.

-p   Parameters needed for the Get.

 One of these optional

  siteId   GUID of the Site to Get.

    Guid[] Guid or SiteId: GUID of the Site to Get.

  siteName   Name of the Site to Get.

    string[] Name or SiteName: Name of the Site to Get.

   Instead of a parameter that matches one of the members listed

    PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

      SiteId

-s   Name of field to sort on. If not included,
                                the first field listed is used. Append a - to
                                the field name to get descending order.

  siteName   Name of the Site. Max Length=50

  description   User description. Default="" Max Length=250

```
-f                              List of fields to retrieve, if not included,
                                all fields are retrieved.

        If only selected fields are needed, pass them in the Fields parameter
                            as a string array.

    siteId                      Read-only GUID that uniquely identifies this
                                Site. Length=36

        Guid or SiteId: Read-only GUID that uniquely identifies this Site.

    siteName                    Name of the Site. Max Length=50

        Name or SiteName: Name of the Site. Max Length=50

    description                 User description. Default="" Max Length=250

        Description: User description. Default="" Max Length=250

    defaultCollectionId         GUID of the Collection to place new Devices
                                into automatically. Not used with
                                defaultCollectionName. Default=null
                                Length=36

        DefaultCollectionId: GUID of the Collection to place new Devices into
                            automatically. Not used with defaultCollectionName.
                            Default=00000000-0000-0000-0000-000000000000

    defaultCollectionName       Name of the Collection to place new Devices
                                into automatically. Not used with
                                defaultCollectionId. Default=null

        DefaultCollectionName: Name of the Collection to place new Devices
                            into automatically. Not used with
                            DefaultCollectionId. Default=""

    inventoryFilePollingInterval  The number of seconds between polls for
                                Disk changes in the Stores. Min=1, Max=600,
                                Default=60

        InventoryFilePollingInterval: The number of seconds between polls for
                            Disk changes in the Stores. Min=1, Max=600,
                            Default=60

    enableDiskUpdate            1 when Disk Updated is enabled for the Site,
                                0 otherwise. Default=0

        EnableDiskUpdate: True when Disk Updated is enabled for the Site,
                            false otherwise. Default=false

    diskUpdateServerId          GUID of the Disk Update Server for the Site.
                                Not used with diskUpdateServerName.
                                Default=null Length=36

        DiskUpdateServerId: GUID of the Disk Update Server for the Site. Not
                            used with DiskUpdateServerName. Default=00000000-
                            0000-0000-0000-000000000000

    diskUpdateServerName        Name of the Disk Update Server for the Site.
                                Not used with diskUpdateServerId.
                                Default=null

        DiskUpdateServerName: Name of the Disk Update Server for the Site.
                            Not used with DiskUpdateServerId. Default=""
```

```
     makUser                        User name used for MAK activation.
                                    Default=null Max Length=64
```

MakUser: User name used for MAK activation. Default="" Max Length=64

```
     makPassword                    User password used for MAK activation.
                                    Default=null Max Length=64
```

MakPassword: User password used for MAK activation. Default="" Max Length=64

```
     role                           Read-only Role of the user for this item.
                                    100 is Farm Administrator, 200 is Site
                                    Administrator, and 999 is read-only.
                                    Default=999
```

Role: Read-only Role of the user for this item. 100 is Farm Administrator, 200 is Site Administrator, and 999 is read-only. Default=999

```
   records                        If successful, records retrieved.
```

PvsSite[]: If successful, the PvsSite object(s) are returned.

Mcli-Get SiteView

```
   Get the fields for a Site View or all Site Views in a Site or the whole
      Farm. All Site Views are returned if no parameters are passed.
```

Get-PvsSiteView

Get the fields for a Site View or all Site Views in a Site or the whole Farm. All Site Views are returned if no parameters are passed.

```
   -p                             Parameters needed for the Get.
    One of these optional
     siteViewId                     GUID of the Site View to Get.
```

Guid[] Guid or SiteViewId: GUID of the Site View to Get.

```
     siteId                         GUID of the Site to Get all Views for.
```

Guid[] SiteId: GUID of the Site to Get all Views for.

```
     siteName                       Name of the Site to Get all Views for.
```

string[] SiteName: Name of the Site to Get all Views for.

```
    or this optional & resolution
     siteViewName                   Name of the Site View to Get.
```

string[] Name or SiteViewName: Name of the Site View to Get.

```
    One of these resolutions when needed
     siteId                         GUID of the Site.
```

Guid[] SiteId: GUID of the Site to Get all Views for.

```
     siteName                       Name of the Site.
```

string[] SiteName: Name of the Site to Get all Views for.

```
     Instead of a parameter that matches one of the members listed
```

```
        PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

          SiteViewId or SiteId
-s                              Name of field to sort on. If not included,
                                the first field listed is used. Append a - to
                                the field name to get descending order.

    siteViewName                Name of the Site View. Max Length=50

    description                 User description. Default="" Max Length=250

    deviceCount                 Read-only count of Devices in this Site
                                 View. Default=0

    deviceWithPVDCount          Read-only count of Devices with Personal
                                 vDisk in this Site View. Default=0

    activeDeviceCount           Read-only count of active Devices in this
                                 Site View. Default=0

-f                              List of fields to retrieve, if not included,
                                all fields are retrieved.

    If only selected fields are needed, pass them in the Fields parameter
                    as a string array.

    siteViewId                  Read-only GUID that uniquely identifies this
                                 Site View. Length=36

      Guid or SiteViewId: Read-only GUID that uniquely identifies this Site
                    View.

    siteViewName                Name of the Site View. Max Length=50

      Name or SiteViewName: Name of the Site View. Max Length=50

    siteId                      GUID of the Site this View is to be a member
                                 of. It is not used with siteName. Length=36

      SiteId: GUID of the Site this View is to be a member of. It is not
                    used with SiteName.

    siteName                    Name of the Site this View is to be a member
                                 of. It is not used with siteId.

      SiteName: Name of the Site this View is to be a member of. It is not
                    used with SiteId.

    description                 User description. Default="" Max Length=250

      Description: User description. Default="" Max Length=250

    deviceCount                 Read-only count of Devices in this Site
                                 View. Default=0

      DeviceCount: Read-only count of Devices in this Site View. Default=0

    deviceWithPVDCount          Read-only count of Devices with Personal
                                 vDisk in this Site View. Default=0

      DeviceWithPVDCount: Read-only count of Devices with Personal vDisk in
                    this Site View. Default=0

    activeDeviceCount           Read-only count of active Devices in this
                                 Site View. Default=0
```

makActivateNeededCount   Read-only count of active Devices that need MAK activation in this Site View. Default=0

role   Read-only Role of the user for this item. 100 is Farm Administrator, and 200 is Site Administrator. Default=999

records   If successful, records retrieved.

PvsSiteView[]: If successful, the PvsSiteView object(s) are returned.

Mcli-Get Store

Get the fields for a Store or all Stores for a Server, Site or the Farm. All Stores are returned if no parameters are passed.

Get-PvsStore

Get the fields for a Store or all Stores for a Server, Site or the Farm. All Stores are returned if no parameters are passed.

-p   Parameters needed for the Get.

One of these optional

storeId   GUID of the Store to Get.

Guid[] Guid or StoreId: GUID of the Store to Get.

storeName   Name of the Store to Get.

string[] Name or StoreName: Name of the Store to Get.

serverId   GUID of the Server to Get all Stores for.

Guid[] ServerId: GUID of the Server to Get all Stores for.

serverName   Name of the Server to Get all Stores for.

string[] ServerName: Name of the Server to Get all Stores for.

siteId   GUID of the Site to Get all Stores for.

Guid[] SiteId: GUID of the Site to Get all Stores for.

siteName   Name of the Site to Get all Stores for.

string[] SiteName: Name of the Site to Get all Stores for.

Instead of a parameter that matches one of the members listed

PvsObject[] Object: PvsObjects with the members below can be used as the Object parameter or from a pileline:

StoreId, ServerId or SiteId

-s   Name of field to sort on. If not included, the first field listed is used. Append a - to the field name to get descending order.

storeName   Name of the Store. Max Length=50

description   User description. Default="" Max Length=250

-f   List of fields to retrieve, if not included,
                             all fields are retrieved.

  If only selected fields are needed, pass them in the Fields parameter
                     as a string array.

storeId   Read-only GUID that uniquely identifies this
                             Store. Length=36

  Guid or StoreId: Read-only GUID that uniquely identifies this Store.

storeName   Name of the Store. Max Length=50

  Name or StoreName: Name of the Store. Max Length=50

siteId   GUID of the Site where Administrators of
                             that Site can change this Store. Not used
                             for Farm Stores. siteName can be used
                             instead. Default=null Length=36

  SiteId: GUID of the Site where Administrators of that Site can change
                     this Store. Not used for Farm Stores. SiteName can
                     be used instead. Default=00000000-0000-0000-0000-
                     000000000000

siteName   Name of the Site where Administrators of
                             that Site can change this Store. Not used
                             for Farm Stores. siteId can be used instead.
                             Default=null

  SiteName: Name of the Site where Administrators of that Site can
                     change this Store. Not used for Farm Stores. SiteId
                     can be used instead. Default=""

description   User description. Default="" Max Length=250

  Description: User description. Default="" Max Length=250

path   Default directory path that the Servers use
                             to access this Store. Max Length=255

  Path: Default directory path that the Servers use to access this
                     Store. Max Length=255

cachePath   Default Cache path(s) that the Servers use
                             with this Store. It is an ordered comma-
                             delimited list. If none are specified the
                             caches will be placed in the WriteCache
                             subdirectory of the Store path. Default=None

  CachePath: Default Cache path(s) that the Servers use with this
                     Store. If none are specified the caches will be
                     placed in the WriteCache subdirectory of the Store
                     path. Default=None

role   Read-only Role of the user for this item.
                             100 is Farm Administrator, 200 is Site
                             Administrator, and 999 is read-only.
                             Default=999

        pathType   Read-only field indicating if the vdisks are
                                      on a server's local hard disk or on a remote
                                      share.

          PathType: Read-only field indicating if the vdisks are on a server's
                     local hard disk or on a remote share.

     records   If successful, records retrieved.

       PvsStore[]: If successful, the PvsStore object(s) are returned.

Mcli-Get StoreSharedOrServerPath

    Get the Stores and paths for the serverName specified or Stores with only
       shared UNC paths.

Get-PvsStoreSharedOrServerPath

    Get the Stores and paths for the ServerName specified or Stores with only
       shared UNC paths.

     -p   Parameters needed for the Get.

      One of these required

       siteId   GUID of a Site used for authorization check.

         Guid[] SiteId: GUID of a Site used for authorization check.

       siteName   Name of a Site used for authorization check.

         string[] SiteName: Name of a Site used for authorization check.

      Optional

       serverName   Name of a Server to also get local Store
                                  paths for.

         string[] ServerName: Name of a Server to also get local Store paths
                      for.

       Instead of a parameter that matches one of the members listed

         PvsObject[] Object: PvsObjects with the members below can be used as
                      the Object parameter or from a pileline:

            SiteId

     -s   Name of field to sort on. If not included,
                                  the first field listed is used. Append a - to
                                  the field name to get descending order.

       storeName   Name of the Store.

     -f   List of fields to retrieve, if not included,
                                  all fields are retrieved.

       If only selected fields are needed, pass them in the Fields parameter
                    as a string array.

       storeId   GUID of the Store.

         StoreId: GUID of the Store.

```
       storeName                     Name of the Store.
```

```
       path                          Directory path that the Servers use to
                                     access this Store.
```

```
    records                          If successful, records retrieved.
```

Mcli-Get Task

Get the current Task fields for select, or all active and completed un-
cleared tasks.

Get-PvsTask

Get the current Task fields for select, or all active and completed un-
cleared tasks.

```
    -p                               Parameters needed for the Get.
     One of these optional
       taskId                        ID of the Task to get.
```

          uint TaskId: ID of the Task to get.

```
       siteId                        GUID of the Site to get Tasks for.
```

          Guid[] SiteId: GUID of the Site to get Tasks for.

```
       siteName                      Name of the Site to get Tasks for.
```

          string[] SiteName: Name of the Site to get Tasks for.

```
     Optional
       state                         The TaskState to get Tasks for. Values are:
                                     0 (Processing), 1 (Cancelled), and 2
                                     (Complete).
```

        uint State: The TaskState to get Tasks for. Values are: 0
                    (Processing), 1 (Cancelled), and 2 (Complete).

```
     Instead of a parameter that matches one of the members listed
```

        PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

          TaskId or SiteId

```
    -s                               Name of field to sort on. If not included,
                                     the first field listed is used. Append a - to
                                     the field name to get descending order.
       taskId                        Unique ID of the task.
    -f                               List of fields to retrieve, if not included,
                                     all fields are retrieved.
```

     If only selected fields are needed, pass them in the Fields parameter
                    as a string array.

```
       taskId                        Unique ID of the task.
```

siteId	GUID of the Site that this Task is being processed in. Default=null Length=36

siteName	Name of the Site that that this Task is being processed in.

handle	Handle to a running function.

serverFqdn	Qualified name of the server. Default=null Max Length=1024

ip	IP Address of the remote host.

port	Port number of the remote service.

startTime	Time the task was started. Format is: YYYY-MM-DD HH:MM:SS.

expirationTime	Time the task record may be removed from the database if the task does not complete. Format is: YYYY-MM-DD HH:MM:SS.

state	State of the Task. Values are: 0 (Processing), 1 (Cancelled), and 2 (Complete). Min=0, Max=2

commandType	Type of the command. Values are: Add, Delete, Get, Info, Run, RunWithReturn, Set and SetList. Default="" Max Length=13

command	Command being processed. Default="" Max Length=50

mapiException	Exception result in XML format. Default=null

```
        results                          Result in XML format. Default=null
```
          Results: Result in XML format. Default=""
```
     records                          If successful, records retrieved.
```
       PvsTask[]: If successful, the PvsTask object(s) are returned.

Mcli-Get UndefinedDisks

```
     Get undefined Disks for the Store on the Server specified.
```

Get-PvsUndefinedDisk

     Get undefined Disks for the Store on the Server specified.

```
      -p                          Parameters needed for the Get.

       One of these required

        serverId                  GUID of the Server to look for undefined
                                  Disks.
```
          Guid[] ServerId: GUID of the Server to look for undefined Disks.
```
        serverName                Name of the Server to look for undefined
                                  Disks.
```
          string[] ServerName: Name of the Server to look for undefined Disks.
```
       One of these required

        storeId                   GUID of the Store that the Server services
                                  to look for undefined Disks.
```
          Guid[] StoreId: GUID of the Store that the Server services to look
                     for undefined Disks.
```
        storeName                 Name of the Store that the Server services
                                  to look for undefined Disks.
```
          string[] StoreName: Name of the Store that the Server services to
                     look for undefined Disks.
```
       Optional

        autoAddEnabled            1 when undefined Disks found should be
                                  automatically added, 0 otherwise. Default=0
```
        SwitchParameter AutoAddEnabled: If -AutoAddEnabled is specified, then
                     undefined Disks found will be automatically added.
```
       Instead of a parameter that matches one of the members listed
```
          PvsObject[] Object: PvsObjects with the members below can be used as
                     the Object parameter or from a pileline:

            ServerId or StoreId
```
      -s                          Name of field to sort on. If not included,
                                  the first field listed is used. Append a - to
                                  the field name to get descending order.

        name                      Name of the disk file without the extension.

      -f                          List of fields to retrieve, if not included,
                                  all fields are retrieved.
```

> If only selected fields are needed, pass them in the Fields parameter as a string array.

name	Name of the disk file without the extension.

> Name: Name of the disk file without the extension.

status	Status of the disk file. Values are: 0 (Valid), 1 (Missing Properties File), 2 (Access Denied), 3 (Access Denied and Missing Properties File), 4 (Invalid Disk File), 5 (Manifest Missing or Invalid), 6 (Both VHD and VHDX)

> Status: Status of the disk file. Values are: 0 (Valid), 1 (Missing Properties File), 2 (Access Denied), 3 (Access Denied and Missing Properties File), 4 (Invalid Disk File), 5 (Manifest Missing or Invalid), 6 (Both VHD and VHDX)

format	Format of the image. Values are: 0 (VHD), 1 (VHDX). Default=0

> VHDX: If VHDX is true, the format of the image is VHDX. Otherwise it is VHD. Default=false

records	If successful, records retrieved.

> PvsUndefinedDisk[]: If successful, the PvsUndefinedDisk object(s) are returned.

Mcli-Get UpdateTask

Get the fields for an Update Task or all Update Tasks in a Site or the whole Farm. All Update Tasks are returned if no parameters are passed.

Get-PvsUpdateTask

> Get the fields for an Update Task or all Update Tasks in a Site or the whole Farm. All Update Tasks are returned if no parameters are passed.

-p	Parameters needed for the Get.

One of these optional

updateTaskId	GUID of the Update Task to Get.

> Guid[] Guid or UpdateTaskId: GUID of the Update Task to Get.

siteId	GUID of the Site to Get all Update Tasks for.

> Guid[] SiteId: GUID of the Site to Get all Update Tasks for.

siteName	Name of the Site to Get all Update Tasks for.

> string[] SiteName: Name of the Site to Get all Update Tasks for.

or this optional & resolution

updateTaskName	Name of the Update Task to Get.

> string[] Name or UpdateTaskName: Name of the Update Task to Get.

One of these resolutions when needed

```
     siteId                        GUID of the Site.
```
   Guid[] SiteId: GUID of the Site to Get all Update Tasks for.
```
     siteName                      Name of the Site.
```
   string[] SiteName: Name of the Site to Get all Update Tasks for.
```
    Instead of a parameter that matches one of the members listed
```
   PvsObject[] Object: PvsObjects with the members below can be used as
                the Object parameter or from a pileline:

     UpdateTaskId or SiteId
```
-s                        Name of field to sort on. If not included,
                          the first field listed is used. Append a - to
                          the field name to get descending order.

   updateTaskName         Name of the Update Task. It is unique within
                          the Site. Max Length=50

   description            User description. Default="" Max Length=250

-f                        List of fields to retrieve, if not included,
                          all fields are retrieved.
```
  If only selected fields are needed, pass them in the Fields parameter
                    as a string array.
```
   updateTaskId           Read-only GUID that uniquely identifies this
                          Update Task. Length=36
```
     Guid or UpdateTaskId: Read-only GUID that uniquely identifies this
                Update Task.
```
   updateTaskName         Name of the Update Task. It is unique within
                          the Site. Max Length=50
```
    Name or UpdateTaskName: Name of the Update Task. It is unique within
                the Site. Max Length=50
```
   siteId                 GUID of the Site that this Update Task is a
                          member of. It is not used with siteName.
                          Length=36
```
     SiteId: GUID of the Site that this Update Task is a member of. It is
                not used with SiteName.
```
   siteName               Name of the Site that this Update Task is a
                          member of. It is not used with siteId.
```
     SiteName: Name of the Site that this Update Task is a member of. It
                is not used with SiteId.
```
   description            User description. Default="" Max Length=250
```
   Description: User description. Default="" Max Length=250
```
   enabled                1 when it will be processed, 0 otherwise.
                          Default=1
```
    Enabled: True when it will be processed, false otherwise.
                Default=true
```
   hour                   The hour of the day to perform the task.
                          Min=0, Max=23, Default=0
```

Hour: The hour of the day to perform the task. Min=0, Max=23, Default=0

minute   The minute of the hour to perform the task. Min=0, Max=59, Default=0

Minute: The minute of the hour to perform the task. Min=0, Max=59, Default=0

recurrence   The update will reoccur on this schedule. 0 = None, 1 = Daily, 2 = Every Weekday, 3 = Weekly, 4 = Monthly Date, 5 = Monthly Type. Min=0, Max=5, Default=0

Recurrence: The update will reoccur on this schedule. 0 = None, 1 = Daily, 2 = Every Weekday, 3 = Weekly, 4 = Monthly Date, 5 = Monthly Type. Min=0, Max=5, Default=0

dayMask   Days selected values. 1 = Monday, 2 = Tuesday, 4 = Wednesday, 8 = Thursday, 16 = Friday, 32 = Saturday, 64 = Sunday, 128 = Day. Default=0. This is used with Weekly and Monthly Type recurrence. Min=1, Max=255, Default=4

DayMask: Days selected values. 1 = Monday, 2 = Tuesday, 4 = Wednesday, 8 = Thursday, 16 = Friday, 32 = Saturday, 64 = Sunday, 128 = Day. Default=0. This is used with Weekly and Monthly Type recurrence. Min=1, Max=255, Default=4

date   Comma delimited days of the month. Numbers from 1-31 are the only valid values. This is used with Monthly Date recurrence. Default="" Max Length=83

Date: Days of the month. Numbers from 1-31 are the only valid values. This is used with Monthly Date recurrence. Default="" Max Length=83

monthlyOffset   When to happen monthly. 0 = None, 1 = First, 2 = Second, 3 = Third, 4 = Forth, 5 = Last. This is used with Monthly Type recurrence. Min=0, Max=5, Default=3

MonthlyOffset: When to happen monthly. 0 = None, 1 = First, 2 = Second, 3 = Third, 4 = Forth, 5 = Last. This is used with Monthly Type recurrence. Min=0, Max=5, Default=3

esdType   Esd to use. Valid values are SCCM or WSUS. If no value, a custom script is run on the client. Default=null Max Length=50

EsdType: Esd to use. Valid values are SCCM or WSUS. If no value, a custom script is run on the client. Default="" Max Length=50

preUpdateScript   Script file to run before the update starts. Default=null Max Length=255

PreUpdateScript: Script file to run before the update starts. Default="" Max Length=255

preVmScript   Script file to run before the VM is loaded.
                            Default=null Max Length=255

  PreVmScript: Script file to run before the VM is loaded. Default=""
                Max Length=255

postUpdateScript   Script file to run after the update
                            finishes. Default=null Max Length=255

  PostUpdateScript: Script file to run after the update finishes.
                Default="" Max Length=255

postVmScript   Script file to run after the VM is unloaded.
                            Default=null Max Length=255

  PostVmScript: Script file to run after the VM is unloaded. Default=""
                Max Length=255

domain   Domain to add the Disk Update Device(s) to.
                            If not included, the first Domain Controller
                            found on the Server is used. Default=null
                            Max Length=255

  Domain: Domain to add the Disk Update Device(s) to. If not included,
                the first Domain Controller found on the Server is
                used. Default="" Max Length=255

organizationUnit   Organizational Unit to add the Disk Update
                            Device(s) to. This parameter is optional. If
                            it is not specified, the device is added to
                            the built in Computers container. Child OU's
                            should be delimited with forward slashes,
                            e.g. "ParentOU/ChildOU". Special characters
                            in an OU name, such as '"', '#', '+', ',',
                            ';', '>', '=', must be escaped with a
                            backslash. For example, an OU called
                            "commaIn,TheMiddle" must be specified as
                            "commaIn\,TheMiddle". The old syntax of
                            delimiting child OU's with a comma is still
                            supported, but deprecated. Note that in this
                            case, the child OU comes first, e.g.
                            "ChildOU,ParentOU". Default=null Max
                            Length=255

  OrganizationUnit: Organizational Unit to add the Disk Update
                Device(s) to. This parameter is optional. If it is
                not specified, the device is added to the built in
                Computers container. Child OU's should be delimited
                with forward slashes, e.g. "ParentOU/ChildOU".
                Special characters in an OU name, such as '"', '#',
                '+', ',', ';', '>', '=', must be escaped with a
                backslash. For example, an OU called
                "commaIn,TheMiddle" must be specified as
                "commaIn\,TheMiddle". The old syntax of delimiting
                child OU's with a comma is still supported, but
                deprecated. Note that in this case, the child OU
                comes first, e.g. "ChildOU,ParentOU". Default="" Max
                Length=255

postUpdateApprove   Access to place the version in after the
                            update has occurred. 0 = Production, 1 =

```
                              Test, 2 = Maintenance. Min=0, Max=2,
                              Default=0
```

PostUpdateApprove: Access to place the version in after the update
has occurred. 0 = Production, 1 = Test, 2 =
Maintenance. Min=0, Max=2, Default=0

```
    records                         If successful, records retrieved.
```

PvsUpdateTask[]: If successful, the PvsUpdateTask object(s) are
returned.

Mcli-Get VirtualHostingPool

```
Get the fields for a Virtual Hosting Pool or all Virtual Hosting Pools in a
   Site or the whole Farm. All Virtual Hosting Pools are returned if no
   parameters are passed.
```

Get-PvsVirtualHostingPool

Get the fields for a Virtual Hosting Pool or all Virtual Hosting Pools in a
Site or the whole Farm. All Virtual Hosting Pools are returned if no
parameters are passed.

```
    -p                              Parameters needed for the Get.
     One of these optional
      virtualHostingPoolId      GUID of the Virtual Hosting Pool to Get.
```

Guid[] Guid or VirtualHostingPoolId: GUID of the Virtual Hosting Pool
to Get.

```
      siteId                        GUID of the Site to Get all Virtual Hosting
                                    Pools for.
```

Guid[] SiteId: GUID of the Site to Get all Virtual Hosting Pools for.

```
      siteName                      Name of the Site to Get all Virtual Hosting
                                    Pools for.
```

string[] SiteName: Name of the Site to Get all Virtual Hosting Pools
for.

```
     or this optional & resolution
      virtualHostingPoolName    Name of the Virtual Hosting Pool to Get.
```

string[] Name or VirtualHostingPoolName: Name of the Virtual Hosting
Pool to Get.

```
     One of these resolutions when needed
      siteId                        GUID of the Site.
```

Guid[] SiteId: GUID of the Site to Get all Virtual Hosting Pools for.

```
      siteName                      Name of the Site.
```

string[] SiteName: Name of the Site to Get all Virtual Hosting Pools
for.

```
      Instead of a parameter that matches one of the members listed
```

PvsObject[] Object: PvsObjects with the members below can be used as
the Object parameter or from a pileline:

VirtualHostingPoolId or SiteId

```
-s                          Name of field to sort on. If not included,
                            the first field listed is used. Append a - to
                            the field name to get descending order.

    virtualHostingPoolName  Name of the Virtual Hosting Pool. It is
                            unique within the Site. Max Length=50

    description             User description. Default="" Max Length=250

-f                          List of fields to retrieve, if not included,
                            all fields are retrieved.

   If only selected fields are needed, pass them in the Fields parameter
                        as a string array.

    virtualHostingPoolId    Read-only GUID that uniquely identifies this
                            Virtual Hosting Pool. Length=36

     Guid or VirtualHostingPoolId: Read-only GUID that uniquely identifies
                    this Virtual Hosting Pool.

    virtualHostingPoolName  Name of the Virtual Hosting Pool. It is
                            unique within the Site. Max Length=50

     Name or VirtualHostingPoolName: Name of the Virtual Hosting Pool. It
                    is unique within the Site. Max Length=50

    siteId                  GUID of the Site that this Virtual Hosting
                            Pool is a member of. It is not used with
                            siteName. Length=36

     SiteId: GUID of the Site that this Virtual Hosting Pool is a member
                    of. It is not used with SiteName.

    siteName                Name of the Site that this Virtual Hosting
                            Pool is a member of. It is not used with
                            siteId.

     SiteName: Name of the Site that this Virtual Hosting Pool is a member
                    of. It is not used with SiteId.

    type                    Type of the Virtual Hosting Pool. 0 = Citrix
                            XenServer, 1 = Microsoft SCVMM/Hyper-V, 2 =
                            VMWare vSphere/ESX. Min=0, Max=3, Default=0

     Type: Type of the Virtual Hosting Pool. 0 = Citrix XenServer, 1 =
                    Microsoft SCVMM/Hyper-V, 2 = VMWare vSphere/ESX.
                    Min=0, Max=3, Default=0

    description             User description. Default="" Max Length=250

     Description: User description. Default="" Max Length=250

    server                  Name or IP of the Host Server. Max
                            Length=255

     Server: Name or IP of the Host Server. Max Length=255

    port                    Port of the Host Server. Min=80, Max=65534,
                            Default=80

     Port: Port of the Host Server. Min=80, Max=65534, Default=80

    datacenter              Datacenter of the Virtual Hosting Pool.
                            Default="" Max Length=250
```

Datacenter: Datacenter of the Virtual Hosting Pool. Default="" Max Length=250

updateLimit   Number of updates at the same time. Min=2, Max=1000, Default=1000

UpdateLimit: Number of updates at the same time. Min=2, Max=1000, Default=1000

updateTimeout   Timeout for updates. Min=2, Max=240, Default=60

UpdateTimeout: Timeout for updates. Min=2, Max=240, Default=60

shutdownTimeout   Timeout for shutdown. Min=2, Max=30, Default=10

ShutdownTimeout: Timeout for shutdown. Min=2, Max=30, Default=10

userName   Name to use when logging into the Server.

UserName: Name to use when logging into the Server.

password   Password to use when logging into the Server.

Password: Password to use when logging into the Server.

xdHostingUnitUuid   UUID of XenDesktop Hosting Unit Default=null Length=36

XdHostingUnitUuid: UUID of XenDesktop Hosting Unit Default=00000000-0000-0000-0000-000000000000

prepopulateEnabled   Enable prepopulate when set to 1 Default=0

PrepopulateEnabled: Enable prepopulate when set to true Default=false

xsPvsSiteUuid   UUID of XenServer PVS_site Default=null Length=36

XsPvsSiteUuid: UUID of XenServer PVS_site Default=00000000-0000-0000-0000-000000000000

platformVersion   Hypervisor Host Version Default=null Max Length=250

PlatformVersion: Hypervisor Host Version Default="" Max Length=250

xdHcHypervisorConnectionName Hypervisor Connection Name for HCL Connection Details object Default=null Max Length=250

XdHcHypervisorConnectionName: Hypervisor Connection Name for HCL Connection Details object Default="" Max Length=250

xdHcHypervisorConnectionUid Hypervisor Connection Uid for HCL Connection Details object Default=null Max Length=250

XdHcHypervisorConnectionUid: Hypervisor Connection Uid for HCL Connection Details object Default="" Max Length=250

xdHcRevision   Revision for HCL Connection Details object Default=null Max Length=250

XdHcRevision: Revision for HCL Connection Details object Default="" Max Length=250

```
        xdHcCustomProperties        Custom Properties for HCL Connection Details
                                     object Default=null Max Length=250

        XdHcCustomProperties: Custom Properties for HCL Connection Details
                           object Default="" Max Length=250

        xdHcSslThumbprints          Ssl Thumbprints for HCL Connection Details
                                     object Default=null Max Length=250

        XdHcSslThumbprints: Ssl Thumbprints for HCL Connection Details object
                           Default="" Max Length=250

    records                         If successful, records retrieved.

       PvsVirtualHostingPool[]: If successful, the PvsVirtualHostingPool
                          object(s) are returned.
```

Mcli-Get XDSite

```
    Get the fields for a XenDesktop Site or all XenDesktop Sites. All
       XenDisktop Sites are returned if no parameters are passed.
```

Get-PvsXDSite

```
    Get the fields for a XenDesktop Site or all XenDesktop Sites. All
       XenDisktop Sites are returned if no parameters are passed.

     -p                             Parameters needed for the Get.

      This optional

       xdSiteId                     GUID of the XenDesktop Site to Get.

         Guid[] Guid or XdSiteId: GUID of the XenDesktop Site to Get.

       Instead of a parameter that matches one of the members listed

         PvsObject[] Object: PvsObjects with the members below can be used as
                         the Object parameter or from a pileline:

          XdSiteId

     -f                             List of fields to retrieve, if not included,
                                     all fields are retrieved.

       If only selected fields are needed, pass them in the Fields parameter
                       as a string array.

       xdSiteId                     GUID of the XenDesktop Site. Length=36

         Guid or XdSiteId: GUID of the XenDesktop Site.

       configServices              Comma delimited list of XenDesktop Server
                                     addresses. Max Length=2000

         ConfigServices: XenDesktop Server addresses. Max Length=2000

    records                         If successful, records retrieved.

       PvsXDSite[]: If successful, the PvsXDSite object(s) are returned.
```

Mcli-Info

```
 Mcli-Info commandName[ -p name=value[, name2=value2]][ -f field[, field2]]
```

Mcli-Info Group

Get the fields for the Groups for the user or the System.

Get-PvsGroup

Get the fields for the Groups for the user or the System.

-p   Parameters needed for the Info request.

 Optional

  domain   Domain of user (may be the name of the local
                            computer).

    string[] Domain: Domain of user (may be the name of the local
                     computer).

  user   Name of user.

    string[] User: Name of user.

  adGroupsEnabled   Get Active Directory groups, when set to 1.
                            Get Windows groups, when set to 0. If not
                            included, the Farm adGroupsEnabled setting
                            is used.

    SwitchParameter AdGroupsEnabled: Get Active Directory groups, when
                     set to true. Get Windows groups, when set to false.
                     If not included, the Farm AdGroupsEnabled setting is
                     used.

-f   List of fields to retrieve, if not included,
                            all fields are retrieved.

   If only selected fields are needed, pass them in the Fields parameter
                  as a string array.

  name   Name of the Group.

    Name: Name of the Group.

  guid   GUID of the Active Directory group. It will
                            be null for Windows groups.

    Guid: GUID of the Active Directory group. 00000000-0000-0000-0000-
                  000000000000 for Windows groups.

 records   If successful, records retrieved.

   PvsGroup[]: If successful, the PvsGroup object(s) are returned.

Mcli-Info LocalServer

Return one record with the local server's NetBios name

Get-PvsLocalServer

Return one record with the local server's NetBios name

-f   List of fields to retrieve, if not included,
                            all fields are retrieved.

  LocalServer   NetBios name of local server.

    Name or LocalServer: NetBios name of local server.

 records   If successful, records retrieved.

PvsLocalServer[]: If successful, the PvsLocalServer object(s) are returned.

Mcli-Info Version

Return one record with the version information.

Get-PvsVersion

Get the version information.

-f   List of fields to retrieve, if not included, all fields are retrieved.

If only selected fields are needed, pass them in the Fields parameter as a string array.

mapiVersion   Version of the system in major.minor.point.build format.

MapiVersion: Version of the system in major.minor.point.build format.

dbVersion   Version of the database schema as a number. Default=0

DbVersion: Version of the database schema as a number. Default=0

type   Type of system. Values are 0 (Normal), 1 (OROM), and 2 (Secure). Default=0

Type: Type of system. Values are 0 (Normal), 1 (OROM), and 2 (Secure). Default=0

dbEdition   Edition of the database. If 'Express Edition', monitor dbSize.

DbEdition: Edition of the database. If 'Express Edition', monitor dbSize.

dbSize   Size of the database in MB. Monitor this value if the edition is 'Express Edition' and this value is close to reaching the 4000 MB maximum. Default=0

DbSize: Size of the database in MB. Monitor this value if the edition is 'Express Edition' and this value is close to reaching the 4000 MB maximum. Default=0

mapiVersionNumber   Internal version number of the system. It is a number that is increaed by 100 for each major and minor release. Point releases are the numbers between each 100. Default=0

MapiVersionNumber: Internal version number of the system. It is a number that is increaed by 100 for each major and minor release. Point releases are the numbers between each 100. Value is 0 when the system does not support MapiVersionNumber. Default=0

records   If successful, records retrieved.

PvsVersion[]: If successful, the PvsVersion object(s) are returned.

Mcli-Run

```
Mcli-Run commandName[ -p name=value[, name2=value2]]
```

Mcli-Run ActivateDevice

Proxy Activate with a Multiple Activation Key and/or apply the Confirmation ID to remote activate a Device DiskLocator pair.

Invoke-PvsActivateDeviceMAK

Proxy Activate with a Multiple Activation Key and/or apply the Confirmation ID to remote activate a Device DiskLocator pair.

-p   Parameters needed for this Run.

 One of these required

  deviceId   GUID of the Device to Activate.

    Guid[] Guid or DeviceId: GUID of the Device to Activate.

  deviceName   Name of the Device to Activate.

    string[] Name or DeviceName: Name of the Device to Activate.

  deviceMac   MAC of the Device to Activate.

    PvsPhysicalAddress[] DeviceMac: MAC of the Device to Activate.

 This required

  makUsedToActivate   Multiple Activation Key to Activate the
                            Device with.

    string[] MakUsedToActivate: Multiple Activation Key to Activate the
                    Device with.

  Instead of a parameter that matches one of the members listed

    PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

     DeviceId

  Optional

    SwitchParameter Confirm: The impact of this operation is "low". If -
                    Confirm is specified, the operation will be
                    confirmed. $ConfirmPreference can be set to "low" to
                    have confirmation without the Confirm parameter.

Mcli-Run AddDeviceToDomain

Assign a Device, all Devices in a Collection or View to a Domain.

Add-PvsDeviceToDomain

Add a Device, all Devices in a Collection or View to a Domain.

-p   Parameters needed for this Run.

 One of these required

  deviceId   GUID of the Device to Add to the Domain.

    Guid[] Guid or DeviceId: GUID of the Device to Add to the Domain.

deviceName   Name of the Device to Add to the Domain.

   string[] Name or DeviceName: Name of the Device to Add to the Domain.

deviceMac   MAC of the Device to Add to the Domain.

   PvsPhysicalAddress[] DeviceMac: MAC of the Device to Add to the Domain.

collectionId   GUID of the Collection to Add all Devices to the Domain.

   Guid[] CollectionId: GUID of the Collection to Add all Devices to the Domain.

siteViewId   GUID of the Site View to Add all Devices to the Domain.

   Guid[] SiteViewId: GUID of the Site View to Add all Devices to the Domain.

farmViewId   GUID of the Farm View to Add all Devices to the Domain.

   Guid[] FarmViewId: GUID of the Farm View to Add all Devices to the Domain.

farmViewName   Name of the Farm View to Add all Devices to the Domain.

   string[] FarmViewName: Name of the Farm View to Add all Devices to the Domain.

or one of these required & resolutions

collectionName   Name of the Collection to Add all Devices to the Domain.

   string[] CollectionName: Name of the Collection to Add all Devices to the Domain.

siteViewName   Name of the Site View to Add all Devices to the Domain.

   string[] SiteViewName: Name of the Site View to Add all Devices to the Domain.

Optional

domain   Domain to add the Device(s) to. If not
                               included, the first Domain Controller found
                               on the Server is used.

   string[] Domain: Domain to add the Device(s) to. If not included, the first Domain Controller found on the Server is used.

organizationUnit   Organizational Unit to add the Device(s) to.
                               This parameter is optional. If it is not
                               specified, the device is added to the built
                               in Computers container. Child OU's should be
                               delimited with forward slashes, e.g.
                               "ParentOU/ChildOU". Special characters in an
                               OU name, such as '"', '#', '+', ',', ';',
                               '>', '=', must be escaped with a backslash.
                               For example, an OU called
                               "commaIn,TheMiddle" must be specified as

"commaIn\,TheMiddle". The old syntax of delimiting child OU's with a comma is still supported, but deprecated. Note that in this case, the child OU comes first, e.g. "ChildOU,ParentOU".

string[] OrganizationUnit: Organizational Unit to add the Device(s) to. This parameter is optional. If it is not specified, the device is added to the built in Computers container. Child OU's should be delimited with forward slashes, e.g. "ParentOU/ChildOU". Special characters in an OU name, such as '"', '#', '+', ',', ';', '>', '=', must be escaped with a backslash. For example, an OU called "commaIn,TheMiddle" must be specified as "commaIn\,TheMiddle". The old syntax of delimiting child OU's with a comma is still supported, but deprecated. Note that in this case, the child OU comes first, e.g. "ChildOU,ParentOU".

One of these resolutions when needed

siteId   GUID of the Site.

  Guid[] SiteId: GUID of the Site.

siteName   Name of the Site.

  string[] SiteName: Name of the Site.

Instead of a parameter that matches one of the members listed

  PvsObject[] Object: PvsObjects with the members below can be used as the Object parameter or from a pileline:

    DeviceId, CollectionId, SiteViewId or FarmViewId

Optional

  SwitchParameter Confirm: The impact of this operation is "low". If -Confirm is specified, the operation will be confirmed. $ConfirmPreference can be set to "low" to have confirmation without the Confirm parameter.

Mcli-Run AddDiskVersion

Add one or more new Versions to a Disk. A manifest file for the new Disk Version(s) must exist in the Store.

Add-PvsDiskVersion

Add one or more new Versions to a Disk. A manifest file for the new Disk Version(s) must exist in the Store.

  -p   Parameters needed for this Run.

  This required

  diskLocatorId   GUID of the Disk Locator File to Add the new Disk Version(s) to.

    Guid[] Guid or DiskLocatorId: GUID of the Disk Locator File to Add the new Disk Version(s) to.

  or this required & resolution

```
          diskLocatorName              Name of the Disk Locator File to Add the new
                                       Disk Version(s) to.

      string[] Name or DiskLocatorName: Name of the Disk Locator File to
                      Add the new Disk Version(s) to.

   One of these resolutions when needed

    siteId                             GUID of the Site.

      Guid[] SiteId: GUID of the Site.

    siteName                           Name of the Site.

      string[] SiteName: Name of the Site.

   One of these resolutions when needed

    storeId                            GUID of the Store that is needed when a
                                       diskLocatorName is used.

      Guid[] StoreId: GUID of the Store that is needed when a
                      DiskLocatorName is used.

    storeName                          Name of the Store that is needed when a
                                       diskLocatorName is used.

      string[] StoreName: Name of the Store that is needed when a
                      DiskLocatorName is used.

   Instead of a parameter that matches one of the members listed

      PvsObject[] Object: PvsObjects with the members below can be used as
                      the Object parameter or from a pileline:

       DiskLocatorId

   Optional

      SwitchParameter Confirm: The impact of this operation is "low". If -
                      Confirm is specified, the operation will be
                      confirmed. $ConfirmPreference can be set to "low" to
                      have confirmation without the Confirm parameter.
```

Mcli-Run ApplyAutoUpdate

```
   Apply Auto Update for a Server or all Servers in a Site.
```

Start-PvsAutoUpdate

```
   Apply Auto Update for a Server or all Servers in a Site.

    -p                                 Parameters needed for this Run.

   One of these required

    serverId                           GUID of the Server to apply Auto Update.

      Guid[] Guid or ServerId: GUID of the Server to apply Auto Update.

    serverName                         Name of the Server to apply Auto Update.

      string[] Name or ServerName: Name of the Server to apply Auto Update.

    siteId                             GUID of the Site to apply Auto Update on all
                                       Servers.

      Guid[] SiteId: GUID of the Site to apply Auto Update on all Servers.
```

```
          siteName                    Name of the Site to apply Auto Update on all
                                       Servers.

           string[] SiteName: Name of the Site to apply Auto Update on all
                         Servers.

        Instead of a parameter that matches one of the members listed

           PvsObject[] Object: PvsObjects with the members below can be used as
                         the Object parameter or from a pileline:

            ServerId or SiteId

         Optional

           SwitchParameter Confirm: The impact of this operation is "low". If -
                         Confirm is specified, the operation will be
                         confirmed. $ConfirmPreference can be set to "low" to
                         have confirmation without the Confirm parameter.
```

Mcli-Run ArchiveAuditTrail

```
   Archive the information in the Audit Trail up to a certain date to a file.
     When finished, the information archived will be removed from the Audit
     Trail.
```

Export-PvsAuditTrail

```
   Archive the information in the Audit Trail up to a certain date to a file.
     When finished, the information archived will be removed from the Audit
     Trail.

     -p                          Parameters needed for this Run.

      This required

       fileName                    Name of the file to archive the Audit Trail
                                    to. This must be a full file path name.

          string[] FileName: Name of the file to archive the Audit Trail to.
                        This must be a full file path name.

       Optional

       endDate                     Last date of information to Archive. If not
                                    entered, all information is Archived. Format
                                    is yyyy/mm/dd.

          DateTime EndDate: Last date of information to Archive. If not
                        entered, all information is Archived. Uses only the
                        date.

       purgeData                   1 when the information archived will be
                                    removed from the Audit Trail. Default=1

          SwitchParameter NoPurgeData: If -NoPurgeData is specified, the
                        information archived will not be removed from the
                        Audit Trail.

       Optional

          SwitchParameter Confirm: The impact of this operation is "low". If -
                        Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "low" to
                        have confirmation without the Confirm parameter.
```

Mcli-Run AssignAuthGroup

Assign an AuthGroup to have Farm, Site or Collection Authorization. If no
   Site or Collection is specified, the AuthGroup is given Farm
   Authorization.

Grant-PvsAuthGroup

Assign an AuthGroup to have Farm, Site or Collection Authorization. If no
   Site or Collection is specified, the AuthGroup is given Farm
   Authorization.

   -p   Parameters needed for this Run.
    One of these required
     authGroupId   GUID of the AuthGroup to assign
                                    Authorization for.

       Guid[] Guid or AuthGroupId: GUID of the AuthGroup to assign
                      Authorization for.

     authGroupName   Name of the AuthGroup to assign
                                    Authorization for.

       string[] Name or AuthGroupName: Name of the AuthGroup to assign
                      Authorization for.

    One of these optional
     siteId   GUID of the Site to assign Authorization to
                                    for the AuthGroup.

       Guid[] SiteId: GUID of the Site to assign Authorization to for the
                      AuthGroup.

     siteName   Name of the Site to assign Authorization to
                                    for the AuthGroup.

       string[] SiteName: Name of the Site to assign Authorization to for
                      the AuthGroup.

     collectionId   GUID of the Collection to assign
                                    Authorization to for the AuthGroup.

       Guid[] CollectionId: GUID of the Collection to assign Authorization
                      to for the AuthGroup.

    or this optional & resolution
     collectionName   Name of the Collection to assign
                                    Authorization to for the AuthGroup.

       string[] CollectionName: Name of the Collection to assign
                      Authorization to for the AuthGroup.

    Optional
     role   Authorization Role for the Collection. 300
                                    or 400 can be used. Role can only be used
                                    with collectionId or collectionName. 300 is
                                    Collection Administrator, and 400 is
                                    Collection Operator. Default=400

       uint Role: Authorization Role for the Collection. 300 or 400 can be
                      used. Role can only be used with CollectionId or

One of these resolutions when needed

siteId   GUID of the Site.

   Guid[] SiteId: GUID of the Site to assign Authorization to for the AuthGroup.

siteName   Name of the Site.

   string[] SiteName: Name of the Site to assign Authorization to for the AuthGroup.

Instead of a parameter that matches one of the members listed

   PvsObject[] Object: PvsObjects with the members below can be used as the Object parameter or from a pileline:

    AuthGroupId, SiteId or CollectionId

Optional

   SwitchParameter Confirm: The impact of this operation is "low". If -Confirm is specified, the operation will be confirmed. $ConfirmPreference can be set to "low" to have confirmation without the Confirm parameter.

Mcli-Run AssignDevice

Assign a Device to a Collection or View. Personal vDisk Devices cannot be moved to another Site.

Move-PvsDeviceToCollection

Move a Device to a Collection. Personal vDisk Devices cannot be moved to a Collection in another Site.

Add-PvsDeviceToView

Add a Device to a View.

  -p   Parameters needed for this Run.

One of these required

deviceId   GUID of the Device to Assign.

   Move-PvsDeviceToCollection Guid[] Guid or DeviceId: GUID of the Device to Move.

   Add-PvsDeviceToView Guid[] Guid or DeviceId: GUID of the Device to Add.

deviceName   Name of the Device to Assign.

   Move-PvsDeviceToCollection string[] Name or DeviceName: Name of the Device to Move.

   Add-PvsDeviceToView string[] Name or DeviceName: Name of the Device to Add.

deviceMac   MAC of the Device to Assign.

   Move-PvsDeviceToCollection PvsPhysicalAddress[] DeviceMac: MAC of the Device to Move.

Add-PvsDeviceToView PvsPhysicalAddress[] DeviceMac: MAC of the Device to Add.

One of these required

  collectionId   GUID of the Collection to Assign a Device or Devices. The Device is moved from whatever Collection it is currently in, to the Collection specified.

Move-PvsDeviceToCollection Guid[] CollectionId: GUID of the Collection to Move a Device to. The Device is moved from whatever Collection it is currently in, to the Collection specified.

  siteViewId   GUID of the Site View to Assign a Device.

Add-PvsDeviceToView Guid[] SiteViewId: GUID of the Site View to Add the Device to.

  farmViewId   GUID of the Farm View to Assign a Device.

Add-PvsDeviceToView Guid[] FarmViewId: GUID of the Farm View to Add the Device to.

  farmViewName   Name of the Farm View to Assign a Device.

Add-PvsDeviceToView string[] FarmViewName: Name of the Farm View to Add the Device to.

or one of these required & resolutions

  collectionName   Name of the Collection to Assign a Device or Devices. The Device is moved from whatever Collection it is currently in, to the Collection specified.

Move-PvsDeviceToCollection string[] CollectionName: Name of the Collection to Move a Device to. The Device is moved from whatever Collection it is currently in, to the Collection specified.

  siteViewName   Name of the Site View to Assign a Device.

Add-PvsDeviceToView string[] SiteViewName: Name of the Site View to Add the Device to.

Optional

  copyTemplate   1 if the Template Device for the collection, if it exists, should be used for the property settings of the assigned Device(s). copyTemplate is only valid with collectionId or collectionName.

Move-PvsDeviceToCollection SwitchParameter CopyTemplate: If -CopyTemplate is specified, the Template Device for the Collection, if it exists, will be used for the property settings of the moved Device.

One of these resolutions when needed

  siteId   GUID of the Site.

Move-PvsDeviceToCollection Guid[] SiteId: GUID of the Site.

Add-PvsDeviceToView Guid[] SiteId: GUID of the Site.

```
     siteName                    Name of the Site.
      Move-PvsDeviceToCollection string[] SiteName: Name of the Site.
      Add-PvsDeviceToView string[] SiteName: Name of the Site.
    Instead of a parameter that matches one of the members listed
      PvsObject[] Object: PvsObjects with the members below can be used as
                     the Object parameter or from a pileline:
        DeviceId, CollectionId, SiteViewId or FarmViewId
    Optional
      SwitchParameter Confirm: The impact of this operation is "low". If -
                     Confirm is specified, the operation will be
                     confirmed. $ConfirmPreference can be set to "low" to
                     have confirmation without the Confirm parameter.
```

Mcli-Run AssignDisk

```
   Assign a Disk to an Update Task.
```

Add-PvsDiskToUpdateTask

```
   Add a Disk to an Update Task.
    -p                           Parameters needed for this Run.
    This required
     diskLocatorId              GUID of the Disk Locator to Assign.
       Guid[] Guid or DiskLocatorId: GUID of the Disk Locator to Assign.
    or this required & resolution
     diskLocatorName            Name of the Disk Locator File to Assign.
       string[] Name or DiskLocatorName: Name of the Disk Locator File to
                     Assign.
    One of these required
     updateTaskId               GUID of the Update Task to Assign a Disk.
       Guid[] UpdateTaskId: GUID of the Update Task to Assign a Disk.
     updateTaskName             Name of the Update Task to Assign a Disk.
       string[] UpdateTaskName: Name of the Update Task to Assign a Disk.
    One of these resolutions when needed
     siteId                     GUID of the Site.
       Guid[] SiteId: GUID of the Site.
     siteName                   Name of the Site.
       string[] SiteName: Name of the Site.
    One of these resolutions when needed
     storeId                    GUID of the Store that is needed when a
                                 diskLocatorName is used.
       Guid[] StoreId: GUID of the Store that is needed when a
                     DiskLocatorName is used.
```

```
storeName                  Name of the Store that is needed when a
                           diskLocatorName is used.
```

string[] StoreName: Name of the Store that is needed when a
                    DiskLocatorName is used.

```
Instead of a parameter that matches one of the members listed
```

PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

  DiskLocatorId or UpdateTaskId

Optional

SwitchParameter Confirm: The impact of this operation is "low". If -
                    Confirm is specified, the operation will be
                    confirmed. $ConfirmPreference can be set to "low" to
                    have confirmation without the Confirm parameter.

Mcli-Run AssignDiskLocator

```
Assign a Disk Locator to a Device, a Collection or View.
```

Add-PvsDiskLocatorToDevice

Assign a Disk Locator to a Device, a Collection or View.

```
 -p                         Parameters needed for this Run.
  This required
  diskLocatorId            GUID of the Disk Locator to Assign.
```

  Guid[] Guid or DiskLocatorId: GUID of the Disk Locator to Assign.

```
  or this required & resolution
  diskLocatorName          Name of the Disk Locator File to Assign.
```

  string[] Name or DiskLocatorName: Name of the Disk Locator File to
                    Assign.

```
  One of these required
  deviceId                 GUID of the Device to Assign a Disk Locator.
```

  Guid[] DeviceId: GUID of the Device to Assign a Disk Locator.

```
  deviceName               Name of the Device to Assign a Disk Locator.
```

  string[] DeviceName: Name of the Device to Assign a Disk Locator.

```
  deviceMac                MAC of the Device to Assign a Disk Locator.
```

  PvsPhysicalAddress[] DeviceMac: MAC of the Device to Assign a Disk
                    Locator.

```
  collectionId             GUID of the Collection to Assign a Disk
                           Locator or Locators to all Devices.
```

  Guid[] CollectionId: GUID of the Collection to Assign a Disk Locator
                    or Locators to all Devices.

```
  siteViewId               GUID of the Site View to Assign a Disk
                           Locator to all Devices.
```

  Guid[] SiteViewId: GUID of the Site View to Assign a Disk Locator to
                    all Devices.

farmViewId   GUID of the Farm View to Assign a Disk
                             Locator to all Devices.

   Guid[] FarmViewId: GUID of the Farm View to Assign a Disk Locator to
                  all Devices.

farmViewName   Name of the Farm View to Assign a Disk
                             Locator to all Devices.

   string[] FarmViewName: Name of the Farm View to Assign a Disk Locator
                  to all Devices.

or one of these required & resolutions

collectionName   Name of the Collection to Assign a Disk
                             Locator or Locators to all Devices.

   string[] CollectionName: Name of the Collection to Assign a Disk
                  Locator or Locators to all Devices.

siteViewName   Name of the Site View to Assign a Disk
                             Locator to all Devices.

   string[] SiteViewName: Name of the Site View to Assign a Disk Locator
                  to all Devices.

Optional

removeExisting   1 to remove the existing Disk Locators
                             before assigning the new one. Default=0

   SwitchParameter RemoveExisting: If -RemoveExisting is specified,
                  remove the existing Disk Locators before assigning
                  the new one.

One of these resolutions when needed

siteId   GUID of the Site.

   Guid[] SiteId: GUID of the Site.

siteName   Name of the Site.

   string[] SiteName: Name of the Site.

One of these resolutions when needed

storeId   GUID of the Store that is needed when a
                             diskLocatorName is used.

   Guid[] StoreId: GUID of the Store that is needed when a
                  DiskLocatorName is used.

storeName   Name of the Store that is needed when a
                             diskLocatorName is used.

   string[] StoreName: Name of the Store that is needed when a
                  DiskLocatorName is used.

 Instead of a parameter that matches one of the members listed

   PvsObject[] Object: PvsObjects with the members below can be used as
                  the Object parameter or from a pileline:

    DiskLocatorId, DeviceId, CollectionId, SiteViewId or FarmViewId

 Optional

Mcli-Run AssignServer

Assign a Server to a Site.

Move-PvsServerToSite

Move a Server to a Site.

```
  -p                            Parameters needed for this Run.
   One of these required
    serverId                    GUID of the Server to Assign.
      Guid[] Guid or ServerId: GUID of the Server to Assign.
    serverName                  Name of the Server to Assign.
      string[] Name or ServerName: Name of the Server to Assign.
   One of these required
    siteId                      GUID of the Site to Assign a Server.
      Guid[] SiteId: GUID of the Site to Assign a Server.
    siteName                    Name of the Site to Assign a Server.
      string[] SiteName: Name of the Site to Assign a Server.
    Instead of a parameter that matches one of the members listed
      PvsObject[] Object: PvsObjects with the members below can be used as
                   the Object parameter or from a pileline:
        ServerId or SiteId
   Optional
      SwitchParameter Confirm: The impact of this operation is "low". If -
                   Confirm is specified, the operation will be
                   confirmed. $ConfirmPreference can be set to "low" to
                   have confirmation without the Confirm parameter.
```

Mcli-Run CancelTask

Cancel a running Task.

Stop-PvsTask

Cancel a running Task.

```
  -p                            Parameters needed for this Run.
   This required
    taskId                      Id of the Task to Cancel.
      uint TaskId: Id of the Task to Cancel.
    Instead of a parameter that matches one of the members listed
      PvsObject[] Object: PvsObjects with the members below can be used as
                   the Object parameter or from a pileline:
```

Mcli-Run ClearTask

Clear a single or all completed or cancelled Tasks in a Site or the whole
Farm.

Clear-PvsTask

Clear a single or all completed or cancelled Tasks in a Site or the whole
Farm.

   -p   Parameters needed for this Run.

    One of these optional

     taskId   Id of the Task to Clear.

       uint TaskId: Id of the Task to Clear.

     siteId   Site Id of the Tasks to Clear.

       Guid[] SiteId: Site Id of the Tasks to Clear.

     siteName   Site Name of the Tasks to Clear.

       string[] SiteName: Site Name of the Tasks to Clear.

     Instead of a parameter that matches one of the members listed

       PvsObject[] Object: PvsObjects with the members below can be used as
                           the Object parameter or from a pileline:

         TaskId or SiteId

      Optional

        SwitchParameter Confirm: The impact of this operation is "medium". If
                        -Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "medium"
                        or "low" to have confirmation without the Confirm
                        parameter.

Mcli-Run CopyPasteDevice

Copy properties of one Device to a Device, all the Devices in a Collection,
Site View or Farm View.

Copy-PvsDeviceProperties

Copy properties of one Device to a Device, all the Devices in a Collection,
Site View or Farm View.

   -p   Parameters needed for this Run.

    One of these required

     deviceIdFrom   GUID of the Device to Copy from.

  Guid DeviceIdFrom: GUID of the Device to Copy from.

deviceNameFrom   Name of the Device to Copy from.

  string DeviceNameFrom: Name of the Device to Copy from.

deviceMacFrom   Mac of the Device to Copy from.

  PvsPhysicalAddress DeviceMacFrom: Mac of the Device to Copy from.

One of these required

deviceId   GUID of the Device to Copy to.

  Guid Guid or DeviceId: GUID of the Device to Copy to.

deviceName   Name of the Device to Copy to.

  string Name or DeviceName: Name of the Device to Copy to.

deviceMac   MAC of the Device to Copy to.

  PvsPhysicalAddress DeviceMac: MAC of the Device to Copy to.

collectionId   GUID of the Collection to Copy to.

  Guid CollectionId: GUID of the Collection to Copy to.

siteViewId   GUID of the Site View to Copy to.

  Guid SiteViewId: GUID of the Site View to Copy to.

farmViewId   GUID of the Farm View to Copy to.

  Guid FarmViewId: GUID of the Farm View to Copy to.

farmViewName   Name of the Farm View to Copy to.

  string FarmViewName: Name of the Farm View to Copy to.

or one of these required & resolutions

collectionName   Name of the Collection to Copy to.

  string CollectionName: Name of the Collection to Copy to.

siteViewName   Name of the Site View to Copy to.

  string SiteViewName: Name of the Site View to Copy to.

Optional

properties   If not specified, all are copied. Comma
                              delimited list of properties. Values are: 1
                              (Description), 2 (Class), 3 (Port), 4
                              (Disabled), 5 (Boot Behavior), 6 (Disk
                              Assignment), 7 (Personality), 8 (Printer), 9
                              (Type), 10 (Authentication) and 11
                              (Logging).

  uint[] Properties: If not specified, all are copied. Properties to
               copy. Values are: 1 (Description), 2 (Class), 3
               (Port), 4 (Disabled), 5 (Boot Behavior), 6 (Disk
               Assignment), 7 (Personality), 8 (Printer), 9 (Type),
               10 (Authentication) and 11 (Logging).

One of these resolutions when needed

siteId   GUID of the Site.

  Guid SiteId: GUID of the Site.

```
     siteName                    Name of the Site.
```

string SiteName: Name of the Site.

```
     Instead of a parameter that matches one of the members listed
```

PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

DeviceId, CollectionId, SiteViewId or FarmViewId

Optional

SwitchParameter Confirm: The impact of this operation is "low". If -
                    Confirm is specified, the operation will be
                    confirmed. $ConfirmPreference can be set to "low" to
                    have confirmation without the Confirm parameter.

Mcli-Run CopyPasteDisk

```
     Copy properties of one Disk to a Disk.
```

Copy-PvsDiskProperties

Copy properties of one Disk to a Disk.

```
     -p                          Parameters needed for this Run.

      This required

      diskLocatorIdFrom       GUID of the Disk Locator to Copy from.
```

Guid DiskLocatorIdFrom: GUID of the Disk Locator to Copy from.

```
      This required

      diskLocatorId           GUID of the Disk Locator to Copy to.
```

Guid Guid or DiskLocatorId: GUID of the Disk Locator to Copy to.

```
      Optional

      properties              If not specified, all are copied. Comma
                              delimited list of properties. Values are: 1
                              (Description), 2 (Class), 3 (Type), 4 (Disk
                              Mode), 5 (Auto Update), 6 (HA), 7 (Active
                              Directory), 8 (Printer), 10 (Version), 11
                              (Date), 12 (Author), 13 (Title), 14
                              (Company), 15 (Internal Filename), 16
                              (Original Filename), 17 (Hardware Target),
                              18 (Menu Text), 19 (Enabled), 20 (Server),
                              and 21 (Store).
```

uint[] Properties: If not specified, all are copied. Properties to
                    copy. Values are: 1 (Description), 2 (Class), 3
                    (Type), 4 (Disk Mode), 5 (Auto Update), 6 (HA), 7
                    (Active Directory), 8 (Printer), 10 (Version), 11
                    (Date), 12 (Author), 13 (Title), 14 (Company), 15
                    (Internal Filename), 16 (Original Filename), 17
                    (Hardware Target), 18 (Menu Text), 19 (Enabled), 20
                    (Server), and 21 (Store).

```
     Instead of a parameter that matches one of the members listed
```

PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

```
            DiskLocatorId

        Optional

          SwitchParameter Confirm: The impact of this operation is "low". If -
                          Confirm is specified, the operation will be
                          confirmed. $ConfirmPreference can be set to "low" to
                          have confirmation without the Confirm parameter.
```

Mcli-Run CopyPasteServer

```
    Copy properties of one Server to a Server.
```

Copy-PvsServerProperties

```
    Copy properties of one Server to a Server.

     -p                          Parameters needed for this Run.
      One of these required

       serverIdFrom              GUID of the Server to Copy from.

         Guid ServerIdFrom: GUID of the Server to Copy from.

       serverNameFrom            Name of the Server to Copy from.

         string ServerNameFrom: Name of the Server to Copy from.

      One of these required

       serverId                  GUID of the Server to Copy to.

         Guid Guid or ServerId: GUID of the Server to Copy to.

       serverName                Name of the Server to Copy to.

         string Name or ServerName: Name of the Server to Copy to.

      Optional

       properties                If not specified, all are copied. Comma
                                 delimited list of properties. Values are: 1
                                 (Configuration), 2 (Port), 4 (Active
                                 Directory), 5 (Advanced Server), 6 (Advanced
                                 Network), 7 (Advanced Pacing), 8 (Advanced
                                 Device) and 9 (Logging).

         uint[] Properties: If not specified, all are copied. Properties to
                          copy. Values are: 1 (Configuration), 2 (Port), 4
                          (Active Directory), 5 (Advanced Server), 6 (Advanced
                          Network), 7 (Advanced Pacing), 8 (Advanced Device)
                          and 9 (Logging).

      Instead of a parameter that matches one of the members listed

         PvsObject[] Object: PvsObjects with the members below can be used as
                          the Object parameter or from a pileline:

           ServerId

        Optional

          SwitchParameter Confirm: The impact of this operation is "low". If -
                          Confirm is specified, the operation will be
                          confirmed. $ConfirmPreference can be set to "low" to
                          have confirmation without the Confirm parameter.
```

Mcli-Run CreateDirectory

Create a Directory on the Server specified.

New-PvsDirectory

Create a Directory on the Server specified.

  -p   Parameters needed for this Run.

  One of these required

   serverId   GUID of the Server to create a Directory on.

    Guid[] Guid or ServerId: GUID of the Server to create a Directory on.

   serverName   Name of the Server to create a Directory on.

    string[] Name or ServerName: Name of the Server to create a Directory on.

  This required

   path   Path of the new Directory to create.

    string[] Path: Path of the new Directory to create.

   Instead of a parameter that matches one of the members listed

    PvsObject[] Object: PvsObjects with the members below can be used as the Object parameter or from a pileline:

     ServerId

  Optional

    SwitchParameter Confirm: The impact of this operation is "low". If -Confirm is specified, the operation will be confirmed. $ConfirmPreference can be set to "low" to have confirmation without the Confirm parameter.

Mcli-Run CreateDiskCancel

Cancel an active CreateDisk.

Stop-PvsCreateDisk

Cancel an active CreateDisk.

  -p   Parameters needed for this Run.

  This required

   name   Name of the Disk file that is being created.

    string[] Name: Name of the Disk file that is being created.

  One of these required

   storeId   GUID of the Store that the Disk will be a member of.

    Guid[] StoreId: GUID of the Store that the Disk will be a member of.

   storeName   Name of the Store that the Disk will be a member of.

    string[] StoreName: Name of the Store that the Disk will be a member of.

```
     Instead of a parameter that matches one of the members listed

       PvsObject[] Object: PvsObjects with the members below can be used as
                          the Object parameter or from a pileline:

          StoreId

       Optional

       SwitchParameter Confirm: The impact of this operation is "medium". If
                          -Confirm is specified, the operation will be
                          confirmed. $ConfirmPreference can be set to "medium"
                          or "low" to have confirmation without the Confirm
                          parameter.
```

Mcli-Run DisableDeviceDiskLocator

```
     Disable a Device's DiskLocator.
```

Disable-PvsDeviceDiskLocator

```
     Disable a Device's DiskLocator.

      -p                         Parameters needed for this Run.

       One of these required

         deviceId                 GUID of the Device to Disable the
                                    DiskLocator for.

          Guid[] Guid or DeviceId: GUID of the Device to Disable the
                          DiskLocator for.

         deviceName               Name of the Device to Disable the
                                    DiskLocator for.

          string[] Name or DeviceName: Name of the Device to Disable the
                          DiskLocator for.

         deviceMac                MAC of the Device to Disable the DiskLocator
                                    for.

          PvsPhysicalAddress[] DeviceMac: MAC of the Device to Disable the
                          DiskLocator for.

       This required

         diskLocatorId            GUID of the DiskLocator to Disable for the
                                    Device.

          Guid[] DiskLocatorId: GUID of the DiskLocator to Disable for the
                          Device.

       or this required & resolution

         diskLocatorName          Name of the DiskLocator File to Disable for
                                    the Device.

          string[] DiskLocatorName: Name of the DiskLocator File to Disable for
                          the Device.

       One of these resolutions when needed

         storeId                  GUID of the Store that is needed when a
                                    diskLocatorName is used.
```

Guid[] StoreId: GUID of the Store that is needed when a
                        DiskLocatorName is used.

        storeName   Name of the Store that is needed when a
                                 diskLocatorName is used.

            string[] StoreName: Name of the Store that is needed when a
                        DiskLocatorName is used.

        Instead of a parameter that matches one of the members listed

            PvsObject[] Object: PvsObjects with the members below can be used as
                        the Object parameter or from a pileline:

            DeviceId or DiskLocatorId

        Optional

            SwitchParameter Confirm: The impact of this operation is "medium". If
                        -Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "medium"
                        or "low" to have confirmation without the Confirm
                        parameter.

Mcli-Run DiskUpdateCancel

    Cancel an active Update Task for an Update Device.

Stop-PvsUpdateTask

    Cancel an active Update Task for an Update Device.

        -p   Parameters needed for this Run.

        This required

        updateTaskId   GUID of the Update Task to Cancel.

            Guid[] UpdateTaskId: GUID of the Update Task to Cancel.

        or this required & resolution

        updateTaskName   Name of the Update Task to Cancel.

            string[] UpdateTaskName: Name of the Update Task to Cancel.

        One of these required

        deviceId   GUID of a specific Update Device to Cancel
                                 an Update Task for.

            Guid[] DeviceId: GUID of a specific Update Device to Cancel an Update
                        Task for.

        deviceName   Name of a specific Update Device to Cancel
                                 an Update Task for.

            string[] DeviceName: Name of a specific Update Device to Cancel an
                        Update Task for.

        One of these resolutions when needed

        siteId   GUID of the Site.

            Guid[] SiteId: GUID of the Site.

        siteName   Name of the Site.

            string[] SiteName: Name of the Site.

```
      Instead of a parameter that matches one of the members listed

       PvsObject[] Object: PvsObjects with the members below can be used as
                        the Object parameter or from a pileline:

        UpdateTaskId or DeviceId

      Optional

       SwitchParameter Confirm: The impact of this operation is "medium". If
                        -Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "medium"
                        or "low" to have confirmation without the Confirm
                        parameter.
```

Mcli-Run DiskUpdateStart

```
   Starts an Update Task.
```

Start-PvsUpdateTask

```
   Starts an Update Task.

    -p                        Parameters needed for this Run.
    This required
     updateTaskId             GUID of the Update Task to Start.
       Guid[] UpdateTaskId: GUID of the Update Task to Start.
    or this required & resolution
     updateTaskName           Name of the Update Task to Start.
       string[] UpdateTaskName: Name of the Update Task to Start.
    One of these resolutions when needed
     siteId                   GUID of the Site.
       Guid[] SiteId: GUID of the Site.
     siteName                 Name of the Site.
       string[] SiteName: Name of the Site.
     Instead of a parameter that matches one of the members listed
       PvsObject[] Object: PvsObjects with the members below can be used as
                        the Object parameter or from a pileline:
        UpdateTaskId
    Optional
       SwitchParameter Confirm: The impact of this operation is "low". If -
                        Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "low" to
                        have confirmation without the Confirm parameter.
```

Mcli-Run EnableDeviceDiskLocator

```
   Enable a Device's DiskLocator. If the DiskLocator is Disabled, that
      overrides the Device DiskLocator setting.
```

Enable-PvsDeviceDiskLocator

Enable a Device's DiskLocator. If the DiskLocator is Disabled, that
  overrides the Device DiskLocator setting.

   -p   Parameters needed for this Run.
    One of these required

     deviceId   GUID of the Device to Enable the DiskLocator
                                for.

       Guid[] Guid or DeviceId: GUID of the Device to Enable the DiskLocator
                     for.

     deviceName   Name of the Device to Enable the DiskLocator
                                for.

       string[] Name or DeviceName: Name of the Device to Enable the
                     DiskLocator for.

     deviceMac   MAC of the Device to Enable the DiskLocator
                                for.

       PvsPhysicalAddress[] DeviceMac: MAC of the Device to Enable the
                     DiskLocator for.

    This required

     diskLocatorId   GUID of the DiskLocator to Enable for the
                                Device.

       Guid[] DiskLocatorId: GUID of the DiskLocator to Enable for the
                     Device.

    or this required & resolution

     diskLocatorName   Name of the DiskLocator to Enable for the
                                Device.

       string[] DiskLocatorName: Name of the DiskLocator to Enable for the
                     Device.

    One of these resolutions when needed

     storeId   GUID of the Store that is needed when a
                                diskLocatorName is used.

       Guid[] StoreId: GUID of the Store that is needed when a
                     DiskLocatorName is used.

     storeName   Name of the Store that is needed when a
                                diskLocatorName is used.

       string[] StoreName: Name of the Store that is needed when a
                     DiskLocatorName is used.

    Instead of a parameter that matches one of the members listed

       PvsObject[] Object: PvsObjects with the members below can be used as
                     the Object parameter or from a pileline:

         DeviceId or DiskLocatorId

     Optional

       SwitchParameter Confirm: The impact of this operation is "low". If -
                     Confirm is specified, the operation will be
                     confirmed. $ConfirmPreference can be set to "low" to
                     have confirmation without the Confirm parameter.

Mcli-Run ExportDisk

Export the disk stack to a manifest file.

Export-PvsDisk

Export the disk stack to a manifest file.

  -p   Parameters needed for this Run.

 This required

  diskLocatorId   GUID of the Disk Locator that identifies the
                            disk to export.

    Guid[] Guid or DiskLocatorId: GUID of the Disk Locator that
               identifies the disk to export.

 or this required & resolution

  diskLocatorName   Name of the Disk Locator that identifies the
                            disk to export.

    string[] Name or DiskLocatorName: Name of the Disk Locator that
               identifies the disk to export.

 Optional

  version   Version to use as the start of the export.
                            The export will include all versions
                            starting with this to the highest one.

    uint Version: Version to use as the start of the export. The export
               will include all versions starting with this to the
               highest one.

 One of these resolutions when needed

  siteId   GUID of the Site.

    Guid[] SiteId: GUID of the Site.

  siteName   Name of the Site.

    string[] SiteName: Name of the Site.

 One of these resolutions when needed

  storeId   GUID of the Store that is needed when a
                          diskLocatorName is used.

    Guid[] StoreId: GUID of the Store that is needed when a
               DiskLocatorName is used.

  storeName   Name of the Store that is needed when a
                          diskLocatorName is used.

    string[] StoreName: Name of the Store that is needed when a
               DiskLocatorName is used.

 Instead of a parameter that matches one of the members listed

    PvsObject[] Object: PvsObjects with the members below can be used as
               the Object parameter or from a pileline:

     DiskLocatorId

Mcli-Run ExportOemLicenses

Oem Only: Export the Oem Licenses for the Devices to the fileName
    specified.

Export-PvsOemLicenses

Oem Only: Export the Oem Licenses for the Devices to the fileName
    specified.

   -p   Parameters needed for this Run.

  This required

    fileName   Name of the file to export the Oem Licenses
                                        to. This must be a full file path name.

      string FileName: Name of the file to export the Oem Licenses to. This
                        must be a full file path name.

  One of these required

    deviceId   GUID of the Device, or comma-delimited list
                                        of Device GUIDs to Export Oem Licenses to
                                        the fileName specified.

      Guid Guid or DeviceId: GUID of the Device to Export Oem Licenses to
                        the fileName specified.

    deviceName   Name of the Device, or comma-delimited list
                                        of Device Names to Export Oem Licenses to
                                        the fileName specified.

      string Name or DeviceName: Name of the Device to Export Oem Licenses
                        to the fileName specified.

    deviceMac   MAC of the Device, or comma-delimited list
                                        of Device MACs to Export Oem Licenses to the
                                        fileName specified.

      PvsPhysicalAddress DeviceMac: MAC of the Device to Export Oem
                        Licenses to the fileName specified.

    collectionId   GUID of the Collection to Export all Device
                                        Oem Licenses to the fileName specified.

      Guid CollectionId: GUID of the Collection to Export all Device Oem
                        Licenses to the fileName specified.

    siteViewId   GUID of the Site View to Export all Device
                                        Oem Licenses to the fileName specified.

      Guid SiteViewId: GUID of the Site View to Export all Device Oem
                        Licenses to the fileName specified.

    farmViewId   GUID of the Farm View to Export all Device
                                        Oem Licenses to the fileName specified.

      Guid FarmViewId: GUID of the Farm View to Export all Device Oem
                        Licenses to the fileName specified.

    farmViewName   Name of the Farm View to Export all Device
                                        Oem Licenses to the fileName specified.

      string FarmViewName: Name of the Farm View to Export all Device Oem
                        Licenses to the fileName specified.

or one of these required & resolutions

  collectionName   Name of the Collection to Export all Device
                                Oem Licenses to the fileName specified.

       string CollectionName: Name of the Collection to Export all Device
                     Oem Licenses to the fileName specified.

  siteViewName   Name of the Site View to Export all Device
                                Oem Licenses to the fileName specified.

       string SiteViewName: Name of the Site View to Export all Device Oem
                   Licenses to the fileName specified.

    One of these resolutions when needed

  siteId   GUID of the Site.

     Guid SiteId: GUID of the Site.

  siteName   Name of the Site.

     string SiteName: Name of the Site.

    Instead of a parameter that matches one of the members listed

     PvsObject[] Object: PvsObjects with the members below can be used as
                   the Object parameter or from a pileline:

       DeviceId, CollectionId, SiteViewId or FarmViewId

Mcli-Run ForceInventory

  Force the Inventory service to refresh its Inventory Table.

Update-PvsInventory

  Force the Inventory service to refresh its Inventory Table.

Mcli-Run ImportDevices

  Import Devices from the contents of the comma or tab delimited fileName
    specified. Each record needs to have Device Name, Mac Address, Site Name,
    Collection Name, optional Description and optional Type. Description must
    exist for Type to be included, but it can have 0 length. Type can be 1
    when it performs test of Disks, 2 when it performs maintenance on Disks,
    and 0 otherwise.

Import-PvsDevices

  Import Devices from the contents of the comma or tab delimited fileName
    specified. Each record needs to have Device Name, Mac Address, Site Name,
    Collection Name, optional Description and optional Type. Description must
    exist for Type to be included, but it can have 0 length. Type can be 1
    when it performs test of Disks, 2 when it performs maintenance on Disks,
    and 0 otherwise.

  -p   Parameters needed for this Run.

   This required

  fileName   Name of the file to import the Devices from.
                                This must be a full file path name.

     string[] Name or FileName: Name of the file to import the Devices
                   from. This must be a full file path name.

One of these optional

collectionId   GUID of the Collection to import the Devices
                                 into.

  Guid[] CollectionId: GUID of the Collection to import the Devices
                 into.

siteId   GUID of the Site to import the Devices into.

  Guid[] SiteId: GUID of the Site to import the Devices into.

siteName   Name of the Site to import the Devices into.

  string[] SiteName: Name of the Site to import the Devices into.

or this optional & resolution

collectionName   Name of the Collection to import the Devices
                                 into.

  string[] CollectionName: Name of the Collection to import the Devices
                 into.

Optional

copyTemplate   1 if the Template Device for the collection,
                                 if it exists, should be used for the
                                 property settings of the imported Devices.

  SwitchParameter CopyTemplate: If -CopyTemplate is specified, the
                 Template Device for the collection, if it exists,
                 will be used for the property settings of the
                 imported Devices.

doNotCreateNewSites   1 if new Sites found in the file should not
                                 be created. Default=0

  SwitchParameter DoNotCreateNewSites: If -DoNotCreateNewSites is
                 specified, new Sites found in the file will not be
                 created.

doNotCreateNewCollections 1 if new Collections found in the file
                                 should not be created. Default=0

  SwitchParameter DoNotCreateNewCollections: If -
                 DoNotCreateNewCollections is specified, new
                 Collections found in the file will not be created.

One of these resolutions when needed

siteId   GUID of the Site.

  Guid[] SiteId: GUID of the Site to import the Devices into.

siteName   Name of the Site.

  string[] SiteName: Name of the Site to import the Devices into.

Instead of a parameter that matches one of the members listed

  PvsObject[] Object: PvsObjects with the members below can be used as
                 the Object parameter or from a pileline:

    CollectionId or SiteId

  Optional

Mcli-Run ImportOemLicenses

Oem Only: Import the Oem Licenses from the contents of the fileName specified.

Import-PvsOemLicenses

Oem Only: Import the Oem Licenses from the contents of the fileName specified.

   -p   Parameters needed for this Run.

  This required

   fileName   Name of the file to import the Oem Licenses from. This must be a full file path name.

    string[] Name or FileName: Name of the file to import the Oem Licenses from. This must be a full file path name.

   Optional

    SwitchParameter Confirm: The impact of this operation is "medium". If -Confirm is specified, the operation will be confirmed. $ConfirmPreference can be set to "medium" or "low" to have confirmation without the Confirm parameter.

Mcli-Run MarkDown

Mark Down a Device, Collection, View, Server or Site.

Invoke-PvsMarkDown

Mark Down a Device, Collection, View, Server or Site.

   -p   Parameters needed for this Run.

  One of these required

   deviceId   GUID of the Device to Mark Down.

    Guid[] DeviceId: GUID of the Device to Mark Down.

   deviceName   Name of the Device to Mark Down.

    string[] DeviceName: Name of the Device to Mark Down.

   deviceMac   MAC of the Device to Mark Down.

    PvsPhysicalAddress[] DeviceMac: MAC of the Device to Mark Down.

   collectionId   GUID of the Collection to Mark Down all Devices.

    Guid[] CollectionId: GUID of the Collection to Mark Down all Devices.

   siteViewId   GUID of the Site View to Mark Down all Devices.

    Guid[] SiteViewId: GUID of the Site View to Mark Down all Devices.

| siteId | GUID of the Site. Can be used alone to Mark Down all Servers and Devices in the Site. |

Guid[] SiteId: GUID of the Site. Can be used alone to Mark Down all Servers and Devices in the Site.

| siteName | Name of the Site. Can be used alone to Mark Down all Servers and Devices in the Site. |

string[] SiteName: Name of the Site. Can be used alone to Mark Down all Servers and Devices in the Site.

| farmViewId | GUID of the Farm View to Mark Down all Devices. |

Guid[] FarmViewId: GUID of the Farm View to Mark Down all Devices.

| farmViewName | Name of the Farm View to Mark Down all Devices. |

string[] FarmViewName: Name of the Farm View to Mark Down all Devices.

| diskLocatorId | GUID of the DiskLocator to Mark Down all Devices. |

Guid[] DiskLocatorId: GUID of the DiskLocator to Mark Down all Devices.

| serverId | GUID of the Server to Mark Down. |

Guid[] ServerId: GUID of the Server to Mark Down.

| serverName | Name of the Server to Mark Down. |

string[] ServerName: Name of the Server to Mark Down.

or one of these required & resolutions

| collectionName | Name of the Collection to Mark Down all Devices. |

string[] CollectionName: Name of the Collection to Mark Down all Devices.

| siteViewName | Name of the Site View to Mark Down all Devices. |

string[] SiteViewName: Name of the Site View to Mark Down all Devices.

| diskLocatorName | Name of the DiskLocator to Mark Down all Devices. |

string[] DiskLocatorName: Name of the DiskLocator to Mark Down all Devices.

One of these resolutions when needed

| siteId | GUID of the Site. |

Guid[] SiteId: GUID of the Site. Can be used alone to Mark Down all Servers and Devices in the Site.

| siteName | Name of the Site. |

string[] SiteName: Name of the Site. Can be used alone to Mark Down all Servers and Devices in the Site.

```
   One of these resolutions when needed

    storeId                        GUID of the Store that is needed when a
                                   diskLocatorName is used.
```

```
    storeName                      Name of the Store that is needed when a
                                   diskLocatorName is used.
```

```
   Instead of a parameter that matches one of the members listed
```

Mcli-Run MergeDisk

```
   Merge the Disk.
```

Merge-PvsDisk

base   When 1, specifies to create a new base from
                            last base plus all updates from that base.
                            The default is to merge all updates from the
                            last base by default.

   SwitchParameter NewBase: If -NewBase is specified, create a new base
                        from last base plus all updates from that base. The
                        default is to merge all updates from the last base
                        by default.

   One of these resolutions when needed

   siteId   GUID of the Site.

      Guid[] SiteId: GUID of the Site.

   siteName   Name of the Site.

      string[] SiteName: Name of the Site.

   One of these resolutions when needed

   storeId   GUID of the Store that is needed when a
                              diskLocatorName is used.

      Guid[] StoreId: GUID of the Store that is needed when a
                        DiskLocatorName is used.

   storeName   Name of the Store that is needed when a
                              diskLocatorName is used.

      string[] StoreName: Name of the Store that is needed when a
                        DiskLocatorName is used.

   Instead of a parameter that matches one of the members listed

      PvsObject[] Object: PvsObjects with the members below can be used as
                        the Object parameter or from a pileline:

        DiskLocatorId

   Optional

      SwitchParameter Confirm: The impact of this operation is "medium". If
                        -Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "medium"
                        or "low" to have confirmation without the Confirm
                        parameter.

Mcli-Run PromoteDiskVersion

   Commit the changes made in the current Maintenance or a Test version.
   Promotes the Maintenance version or a Test version to a Test or new
   Production version.

Invoke-PvsPromoteDiskVersion

   Commit the changes made in the current Maintenance or a Test version.
   Promotes the Maintenance version or a Test version to a Test or new
   Production version.

   -p   Parameters needed for this Run.

      This required

```
   diskLocatorId              GUID of the Disk Locator File to Promote the
                              Disk Version of.

   Guid[] Guid or DiskLocatorId: GUID of the Disk Locator File to
                   Promote the Disk Version of.
or this required & resolution

   diskLocatorName            Name of the Disk Locator File to Promote the
                              Disk Version of.

   string[] Name or DiskLocatorName: Name of the Disk Locator File to
                   Promote the Disk Version of.
Optional

   scheduledDate              Date/Time the new disk version will become
                              available. Format is: YYYY-MM-DD HH:MM

   DateTime ScheduledDate: Date/Time the new disk version will become
                   available. Uses only the date, hour and minute.

   testVersion                Specifies the Test version number that
                              should be Promoted to Production.

   uint TestVersion: Specifies the Test version number that should be
                   Promoted to Production.

   test                       Set the mode of the Maintenance version to
                              Test when 1. Default is 0.

   SwitchParameter Test: If -Test is specified, set the mode of the
                   Maintenance version to Test.
One of these resolutions when needed

  siteId                      GUID of the Site.

   Guid[] SiteId: GUID of the Site.

  siteName                    Name of the Site.

   string[] SiteName: Name of the Site.
One of these resolutions when needed

  storeId                     GUID of the Store that is needed when a
                              diskLocatorName is used.

   Guid[] StoreId: GUID of the Store that is needed when a
                   DiskLocatorName is used.

  storeName                   Name of the Store that is needed when a
                              diskLocatorName is used.

   string[] StoreName: Name of the Store that is needed when a
                   DiskLocatorName is used.
  Instead of a parameter that matches one of the members listed

   PvsObject[] Object: PvsObjects with the members below can be used as
                   the Object parameter or from a pileline:

    DiskLocatorId

  Optional

   SwitchParameter Confirm: The impact of this operation is "low". If -
                   Confirm is specified, the operation will be
```

Mcli-Run RemoveAuthGroup

Remove Farm, Site or Collection Authorization for an AuthGroup. If no Site
or Collection is specified, Farm Authorization is removed for the
AuthGroup.

Revoke-PvsAuthGroup

Remove Farm, Site or Collection Authorization for an AuthGroup. If no Site
or Collection is specified, Farm Authorization is removed for the
AuthGroup.

-p   Parameters needed for this Run.
 One of these required

  authGroupId   GUID of the AuthGroup to remove
                             Authorization for.

    Guid[] Guid or AuthGroupId: GUID of the AuthGroup to remove
                Authorization for.

  authGroupName   Name of the AuthGroup to remove
                             Authorization for.

    string[] Name or AuthGroupName: Name of the AuthGroup to remove
                Authorization for.

 One of these optional

  siteId   GUID of the Site to remove Authorization for
                             the AuthGroup.

    Guid[] SiteId: GUID of the Site to remove Authorization for the
                AuthGroup.

  siteName   Name of the Site to remove Authorization for
                             the AuthGroup.

    string[] SiteName: Name of the Site to remove Authorization for the
                AuthGroup.

  collectionId   GUID of the Collection to remove
                             Authorization for the AuthGroup.

    Guid[] CollectionId: GUID of the Collection to remove Authorization
                for the AuthGroup.

 or this optional & resolution

  collectionName   Name of the Collection to remove
                             Authorization for the AuthGroup.

    string[] CollectionName: Name of the Collection to remove
                Authorization for the AuthGroup.

 One of these resolutions when needed

  siteId   GUID of the Site.

    Guid[] SiteId: GUID of the Site to remove Authorization for the
                AuthGroup.

  siteName   Name of the Site.

Instead of a parameter that matches one of the members listed

Mcli-Run RemoveDevice

Remove a Device from a View.

Remove-PvsDeviceFromView

Remove a Device from a View.

   -p   Parameters needed for this Run.

 One of these required

  deviceId   GUID of the Device to Remove.

    Guid[] Guid or DeviceId: GUID of the Device to Remove.

  deviceName   Name of the Device to Remove.

    string[] Name or DeviceName: Name of the Device to Remove.

  deviceMac   MAC of the Device to Remove.

    PvsPhysicalAddress[] DeviceMac: MAC of the Device to Remove.

 One of these required

  siteViewId   GUID of the Site View to Remove the Devices
                                 from.

    Guid[] SiteViewId: GUID of the Site View to Remove the Devices from.

  farmViewId   GUID of the Farm View to Remove the Devices
                                 from.

    Guid[] FarmViewId: GUID of the Farm View to Remove the Devices from.

  farmViewName   Name of the Farm View to Remove the Devices
                                 from.

    string[] FarmViewName: Name of the Farm View to Remove the Devices
                    from.

 or this required & resolution

  siteViewName   Name of the Site View to Remove the Devices
                                 from.

    string[] SiteViewName: Name of the Site View to Remove the Devices
                    from.

 One of these resolutions when needed

```
        siteId                      GUID of the Site.

         Guid[] SiteId: GUID of the Site.

        siteName                    Name of the Site.

         string[] SiteName: Name of the Site.

        Instead of a parameter that matches one of the members listed

         PvsObject[] Object: PvsObjects with the members below can be used as
                      the Object parameter or from a pileline:

          DeviceId, SiteViewId or FarmViewId

        Optional

         SwitchParameter Confirm: The impact of this operation is "low". If -
                      Confirm is specified, the operation will be
                      confirmed. $ConfirmPreference can be set to "low" to
                      have confirmation without the Confirm parameter.
```

Mcli-Run RemoveDeviceFromDomain

```
        Remove a Device, all Devices in a Collection or View from a Domain.
```

Remove-PvsDeviceFromDomain

```
        Remove a Device, all Devices in a Collection or View from a Domain.

         -p                         Parameters needed for this Run.

          One of these required

         deviceId                    GUID of the Device to Remove from the
                                     Domain.

           Guid[] Guid or DeviceId: GUID of the Device to Remove from the
                        Domain.

         deviceName                  Name of the Device to Remove from the
                                     Domain.

           string[] Name or DeviceName: Name of the Device to Remove from the
                        Domain.

         deviceMac                  MAC of the Device to Remove from the Domain.

           PvsPhysicalAddress[] DeviceMac: MAC of the Device to Remove from the
                        Domain.

         collectionId                GUID of the Collection to Remove all Devices
                                     from the Domain.

           Guid[] CollectionId: GUID of the Collection to Remove all Devices
                        from the Domain.

         siteViewId                  GUID of the Site View to Remove all Devices
                                     from the Domain.

           Guid[] SiteViewId: GUID of the Site View to Remove all Devices from
                        the Domain.

         farmViewId                  GUID of the Farm View to Remove all Devices
                                     from the Domain.

           Guid[] FarmViewId: GUID of the Farm View to Remove all Devices from
                        the Domain.
```

```
      farmViewName                    Name of the Farm View toRemove all Devices
                                      from the Domain.
```

```
       or one of these required & resolutions

      collectionName                  Name of the Collection to Remove all Devices
                                      from the Domain.
```

```
      siteViewName                    Name of the Site View to Remove all Devices
                                      from the Domain.
```

```
   Optional

      domain                          Domain to remove the Device(s) from. If not
                                      included, the first Domain Controller found
                                      on the Server is used.
```

```
   One of these resolutions when needed

     siteId                           GUID of the Site.
```

```
       Guid[] SiteId: GUID of the Site.
```

```
     siteName                         Name of the Site.
```

```
       string[] SiteName: Name of the Site.
```

```
   Instead of a parameter that matches one of the members listed

       PvsObject[] Object: PvsObjects with the members below can be used as
                      the Object parameter or from a pileline:

        DeviceId, CollectionId, SiteViewId or FarmViewId

   Optional

       SwitchParameter Confirm: The impact of this operation is "medium". If
                      -Confirm is specified, the operation will be
                      confirmed. $ConfirmPreference can be set to "medium"
                      or "low" to have confirmation without the Confirm
                      parameter.
```

Mcli-Run RemoveDirectory

```
   Remove a Directory on the Server specified.
```

Remove-PvsDirectory

```
   Remove a Directory on the Server specified.

    -p                               Parameters needed for this Run.

     One of these required
```

```
         serverId                  GUID of the Server to remove a Directory
                                    from. The directory must be empty to be
                                    removed.
```
         Guid[] Guid or ServerId: GUID of the Server to remove a Directory
                         from. The directory must be empty to be removed.
```
         serverName                Name of the Server to remove a Directory
                                    from. The directory must be empty to be
                                    removed.
```
         string[] Name or ServerName: Name of the Server to remove a Directory
                         from. The directory must be empty to be removed.
```
      This required
```
```
       path                        Path of the Directory to remove.
```
         string[] Path: Path of the Directory to remove.
```
       Instead of a parameter that matches one of the members listed
```
         PvsObject[] Object: PvsObjects with the members below can be used as
                         the Object parameter or from a pileline:
```
           ServerId
```
      Optional
```
         SwitchParameter Confirm: The impact of this operation is "medium". If
                         -Confirm is specified, the operation will be
                         confirmed. $ConfirmPreference can be set to "medium"
                         or "low" to have confirmation without the Confirm
                         parameter.

## Mcli-Run RemoveDisk

```
   Remove a Disk from an Update Task.
```

Remove-PvsDiskFromUpdateTask

   Remove a Disk from an Update Task.

```
    -p                              Parameters needed for this Run.
```
```
     This required
```
```
      diskLocatorId              GUID of the Disk Locator to Remove.
```
        Guid[] Guid or DiskLocatorId: GUID of the Disk Locator to Remove.
```
     or this required & resolution
```
```
      diskLocatorName            Name of the Disk Locator File to Remove.
```
        string[] Name or DiskLocatorName: Name of the Disk Locator File to
                        Remove.
```
     One of these optional
```
```
      updateTaskId               GUID of the Update Task to Remove a Disk.
```
        Guid[] UpdateTaskId: GUID of the Update Task to Remove a Disk.
```
      updateTaskName             Name of the Update Task to Remove a Disk.
```
        string[] UpdateTaskName: Name of the Update Task to Remove a Disk.
```
     One of these resolutions when needed
```

```
        siteId                       GUID of the Site.

         Guid[] SiteId: GUID of the Site.

        siteName                     Name of the Site.

         string[] SiteName: Name of the Site.

     One of these resolutions when needed

        storeId                      GUID of the Store that is needed when a
                                     diskLocatorName is used.

         Guid[] StoreId: GUID of the Store that is needed when a
                       DiskLocatorName is used.

        storeName                    Name of the Store that is needed when a
                                     diskLocatorName is used.

         string[] StoreName: Name of the Store that is needed when a
                       DiskLocatorName is used.

     Instead of a parameter that matches one of the members listed

         PvsObject[] Object: PvsObjects with the members below can be used as
                       the Object parameter or from a pileline:

           DiskLocatorId or UpdateTaskId

      Optional

         SwitchParameter Confirm: The impact of this operation is "medium". If
                       -Confirm is specified, the operation will be
                       confirmed. $ConfirmPreference can be set to "medium"
                       or "low" to have confirmation without the Confirm
                       parameter.
```

Mcli-Run RemoveDiskLocator

```
     Remove a Disk Locator from a Device, Collection, View, or Site.
```

Remove-PvsDiskLocatorFromDevice

```
     Remove a Disk Locator from a Device, Collection, View, or Site.

       -p                           Parameters needed for this Run.

        This required

        diskLocatorId                GUID of the Disk Locator to Remove.

          Guid[] Guid or DiskLocatorId: GUID of the Disk Locator to Remove.

       or this required & resolution

        diskLocatorName              Name of the Disk Locator File to Remove.

          string[] Name or DiskLocatorName: Name of the Disk Locator File to
                       Remove.

       One of these optional

        deviceId                     GUID of the Device to Remove a Disk Locator.

          Guid[] DeviceId: GUID of the Device to Remove a Disk Locator.

        deviceName                   Name of the Device to Remove a Disk Locator.

          string[] DeviceName: Name of the Device to Remove a Disk Locator.
```

deviceMac   MAC of the Device to Remove a Disk Locator.

  PvsPhysicalAddress[] DeviceMac: MAC of the Device to Remove a Disk
                  Locator.
collectionId   GUID of the Collection to Remove a Disk
                                Locator or Locators from all Devices.

  Guid[] CollectionId: GUID of the Collection to Remove a Disk Locator
                  or Locators from all Devices.
siteViewId   GUID of the Site View to Remove a Disk
                                Locator from all Devices.

  Guid[] SiteViewId: GUID of the Site View to Remove a Disk Locator
                  from all Devices.
farmViewId   GUID of the Farm View to Remove a Disk
                                Locator from all Devices.

  Guid[] FarmViewId: GUID of the Farm View to Remove a Disk Locator
                  from all Devices.
farmViewName   Name of the Farm View to Remove a Disk
                                Locator from all Devices.

  string[] FarmViewName: Name of the Farm View to Remove a Disk Locator
                  from all Devices.

or one of these optional & resolutions

collectionName   Name of the Collection to Remove a Disk
                                Locator or Locators from all Devices.

  string[] CollectionName: Name of the Collection to Remove a Disk
                  Locator or Locators from all Devices.
siteViewName   Name of the Site View to Remove a Disk
                                Locator from all Devices.

  string[] SiteViewName: Name of the Site View to Remove a Disk Locator
                  from all Devices.

One of these resolutions when needed

siteId   GUID of the Site.

  Guid[] SiteId: GUID of the Site.
siteName   Name of the Site.

  string[] SiteName: Name of the Site.

One of these resolutions when needed

storeId   GUID of the Store that is needed when a
                                diskLocatorName is used.

  Guid[] StoreId: GUID of the Store that is needed when a
                  DiskLocatorName is used.
storeName   Name of the Store that is needed when a
                                diskLocatorName is used.

  string[] StoreName: Name of the Store that is needed when a
                  DiskLocatorName is used.

  Instead of a parameter that matches one of the members listed

PvsObject[] Object: PvsObjects with the members below can be used as
                        the Object parameter or from a pileline:

        DiskLocatorId, DeviceId, CollectionId, SiteViewId or FarmViewId

   Optional

        SwitchParameter Confirm: The impact of this operation is "medium". If
                        -Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "medium"
                        or "low" to have confirmation without the Confirm
                        parameter.

Mcli-Run ResetDatabaseConnection

        Cause the database location to be reloaded.

Reset-PvsDatabase

        Cause the database location to be reloaded.

Mcli-Run ResetDeviceForDomain

        Reset a Device, all Devices in a Collection or View for a Domain.

Reset-PvsDeviceForDomain

        Reset a Device, all Devices in a Collection or View for a Domain.

     -p   Parameters needed for this Run.

      One of these required

       deviceId   GUID of the Device to Reset for the Domain.

        Guid[] Guid or DeviceId: GUID of the Device to Reset for the Domain.

       deviceName   Name of the Device to Reset for the Domain.

        string[] Name or DeviceName: Name of the Device to Reset for the
                        Domain.

       deviceMac   MAC of the Device to Reset for the Domain.

        PvsPhysicalAddress[] DeviceMac: MAC of the Device to Reset for the
                        Domain.

       collectionId   GUID of the Collection to Reset all Devices
                                  for the Domain.

        Guid[] CollectionId: GUID of the Collection to Reset all Devices for
                        the Domain.

       siteViewId   GUID of the Site View to Reset all Devices
                                  for the Domain.

        Guid[] SiteViewId: GUID of the Site View to Reset all Devices for the
                        Domain.

       farmViewId   GUID of the Farm View to Reset all Devices
                                  for the Domain.

        Guid[] FarmViewId: GUID of the Farm View to Reset all Devices for the
                        Domain.

       farmViewName   Name of the Farm View to Reset all Devices
                                  for the Domain.

```
            string[] FarmViewName: Name of the Farm View to Reset all Devices for
                      the Domain.
   or one of these required & resolutions
    collectionName              Name of the Collection to Reset all Devices
                                for the Domain.

            string[] CollectionName: Name of the Collection to Reset all Devices
                      for the Domain.

    siteViewName                Name of the Site View to Reset all Devices
                                for the Domain.

            string[] SiteViewName: Name of the Site View to Reset all Devices for
                      the Domain.
   Optional
    domain                      Domain to Reset the Device(s) for. If not
                                included, the first Domain Controller found
                                on the Server is used.

        string[] Domain: Domain to Reset the Device(s) for. If not included,
                      the first Domain Controller found on the Server is
                      used.

    organizationUnit            Organizational Unit to reset the Device(s)
                                to. This parameter is optional. If it is not
                                specified, the account remains in its
                                existing OU. Child OU's should be delimited
                                with forward slashes, e.g.
                                "ParentOU/ChildOU". Special characters in an
                                OU name, such as '"', '#', '+', ',', ';',
                                '>', '=', must be escaped with a backslash.
                                For example, an OU called
                                "commaIn,TheMiddle" must be specified as
                                "commaIn\,TheMiddle". The old syntax of
                                delimiting child OU's with a comma is still
                                supported, but deprecated. Note that in this
                                case, the child OU comes first, e.g.
                                "ChildOU,ParentOU".

        string[] OrganizationUnit: Organizational Unit to reset the Device(s)
                      to. This parameter is optional. If it is not
                      specified, the account remains in its existing OU.
                      Child OU's should be delimited with forward slashes,
                      e.g. "ParentOU/ChildOU". Special characters in an OU
                      name, such as '"', '#', '+', ',', ';', '>', '=',
                      must be escaped with a backslash. For example, an OU
                      called "commaIn,TheMiddle" must be specified as
                      "commaIn\,TheMiddle". The old syntax of delimiting
                      child OU's with a comma is still supported, but
                      deprecated. Note that in this case, the child OU
                      comes first, e.g. "ChildOU,ParentOU".
   One of these resolutions when needed
    siteId                      GUID of the Site.

      Guid[] SiteId: GUID of the Site.

    siteName                    Name of the Site.
```

```
            string[] SiteName: Name of the Site.
        Instead of a parameter that matches one of the members listed

          PvsObject[] Object: PvsObjects with the members below can be used as
                        the Object parameter or from a pileline:

            DeviceId, CollectionId, SiteViewId or FarmViewId

        Optional

          SwitchParameter Confirm: The impact of this operation is "medium". If
                        -Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "medium"
                        or "low" to have confirmation without the Confirm
                        parameter.
```

Mcli-Run RestartStreamService

```
    Restart the Stream Service on a Server or all Servers in a Site.
```

Restart-PvsStreamService

```
    Restart the Stream Service on a Server or all Servers in a Site.

      -p                          Parameters needed for this Run.
       One of these required

        serverId                  GUID of the Server to restart the Stream
                                  Service.

          Guid[] Guid or ServerId: GUID of the Server to restart the Stream
                        Service.

        serverName                Name of the Server to restart the Stream
                                  Service.

          string[] Name or ServerName: Name of the Server to restart the Stream
                        Service.

        siteId                    GUID of the Site to restart the Stream
                                  Service on all Servers.

          Guid[] SiteId: GUID of the Site to restart the Stream Service on all
                        Servers.

        siteName                  Name of the Site to restart the Stream
                                  Service on all Servers.

          string[] SiteName: Name of the Site to restart the Stream Service on
                        all Servers.

        Instead of a parameter that matches one of the members listed

          PvsObject[] Object: PvsObjects with the members below can be used as
                        the Object parameter or from a pileline:

            ServerId or SiteId

        Optional

          SwitchParameter Confirm: The impact of this operation is "medium". If
                        -Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "medium"
                        or "low" to have confirmation without the Confirm
                        parameter.
```

Mcli-Run RevertDiskVersion

Set the existing highest version disk to Maintenance or Test mode. A
   specified version can be reverted to Test mode if there are no Production
   versions higher than it. If the mode is Test, it can be set to
   Maintenance.

Invoke-PvsRevertDiskVersion

Set the existing highest version disk to Maintenance or Test mode. A
   specified version can be reverted to Test mode if there are no Production
   versions higher than it. If the mode is Test, it can be set to
   Maintenance.

    -p   Parameters needed for this Run.

     This required

      diskLocatorId   GUID of the Disk Locator File to Revert.

        Guid[] Guid or DiskLocatorId: GUID of the Disk Locator File to
                     Revert.

      or this required & resolution

      diskLocatorName   Name of the Disk Locator File to Revert.

        string[] Name or DiskLocatorName: Name of the Disk Locator File to
                     Revert.

     Optional

      version   Specifies the version number that should be
                                Reverted to Test mode.

        uint Version: Specifies the version number that should be Reverted to
                     Test mode.

      test   If reverting the highest version and 1 then
                                set the access to Test, otherwise set it to
                                Maintenance.

        SwitchParameter Test: If -Test is specified, when reverting the
                     highest version, the access will be set to Test,
                     otherwise set it to Maintenance.

     One of these resolutions when needed

      siteId   GUID of the Site.

        Guid[] SiteId: GUID of the Site.

      siteName   Name of the Site.

        string[] SiteName: Name of the Site.

     One of these resolutions when needed

      storeId   GUID of the Store that is needed when a
                                diskLocatorName is used.

        Guid[] StoreId: GUID of the Store that is needed when a
                     DiskLocatorName is used.

      storeName   Name of the Store that is needed when a
                                diskLocatorName is used.

       Instead of a parameter that matches one of the members listed

         PvsObject[] Object: PvsObjects with the members below can be used as
                     the Object parameter or from a pileline:

           DiskLocatorId

       Optional

         SwitchParameter Confirm: The impact of this operation is "medium". If
                     -Confirm is specified, the operation will be
                     confirmed. $ConfirmPreference can be set to "medium"
                     or "low" to have confirmation without the Confirm
                     parameter.

Mcli-Run SetOverrideVersion

Specify a Disk Version all Production Devices will boot from.

Set-PvsOverrideVersion

Specify a Disk Version all Production Devices will boot from.

   -p   Parameters needed for this Run.

    This required

     diskLocatorId   GUID of the Disk Locator File to Override
                             the Production Version for.

        Guid[] Guid or DiskLocatorId: GUID of the Disk Locator File to
                     Override the Production Version for.

     or this required & resolution

     diskLocatorName   Name of the Disk Locator File to Override
                             the Production Version for.

        string[] Name or DiskLocatorName: Name of the Disk Locator File to
                     Override the Production Version for.

     Optional

      version   Version to set as the Override Production
                              Version. If Version is not included and if
                              there is an Override Production Version,
                              then no longer have it as the Override
                              Version.

        uint Version: Version to set as the Override Production Version. If
                     Version is not included and if there is an Override
                     Production Version, then no longer have it as the
                     Override Version.

    One of these resolutions when needed

     siteId   GUID of the Site.

        Guid[] SiteId: GUID of the Site.

     siteName   Name of the Site.

        string[] SiteName: Name of the Site.

One of these resolutions when needed

storeId   GUID of the Store that is needed when a
                                diskLocatorName is used.

   Guid[] StoreId: GUID of the Store that is needed when a
                     DiskLocatorName is used.

storeName   Name of the Store that is needed when a
                                diskLocatorName is used.

   string[] StoreName: Name of the Store that is needed when a
                     DiskLocatorName is used.

Instead of a parameter that matches one of the members listed

   PvsObject[] Object: PvsObjects with the members below can be used as
                     the Object parameter or from a pileline:

     DiskLocatorId

Optional

   SwitchParameter Confirm: The impact of this operation is "low". If -
                     Confirm is specified, the operation will be
                     confirmed. $ConfirmPreference can be set to "low" to
                     have confirmation without the Confirm parameter.

Mcli-Run StartDeviceDiskTempVersionMode

Set a Temporary Disk Version for the specified Device and optional
   DiskLocator. The Temporary Disk Version uses the production version that
   all production Devices boot from. Once set, the Device boots this
   Temporary Disk Version instead of any currently assigned vDisk. Not
   supported for non-production and Personal vDisk Devices. Cannot be done
   when the the Device already has a Temporary Disk Version, when the
   DiskLocator is using server side persistent cache mode or the active
   production version is in private mode.

Start-PvsDeviceDiskTempVersionMode

Set a Temporary Disk Version for the specified Device and optional
   DiskLocator. The Temporary Disk Version uses the production version that
   all production Devices boot from. Once set, the Device boots this
   Temporary Disk Version instead of any currently assigned vDisk. Not
   supported for non-production and Personal vDisk Devices. Cannot be done
   when the the Device already has a Temporary Disk Version, when the
   DiskLocator is using server side persistent cache mode or the active
   production version is in private mode.

-p   Parameters needed for this Run.

   One of these required

deviceId   GUID of the Device to start to use a
                                temporary disk version.

   Guid[] Guid or DeviceId: GUID of the Device to start to use a
                     temporary disk version.

deviceName   Name of the Device to start to use a
                                temporary disk version.

   string[] Name or DeviceName: Name of the Device to start to use a
                     temporary disk version.

```
        deviceMac                  MAC of the Device to start to use a
                                   temporary disk version.

        PvsPhysicalAddress[] DeviceMac: MAC of the Device to start to use a
                      temporary disk version.

     This optional

        diskLocatorId              GUID of the Disk Locator to use a temporary
                                   version of. If not specified, the only Disk
                                   Locator assigned to the Device is used.

        Guid[] DiskLocatorId: GUID of the Disk Locator to use a temporary
                      version of. If not specified, the only Disk Locator
                      assigned to the Device is used.

     or this optional & resolution

        diskLocatorName            Name of the Disk Locator to use a temporary
                                   version of. If not specified, the only Disk
                                   Locator assigned to the Device is used.

        string[] DiskLocatorName: Name of the Disk Locator to use a temporary
                      version of. If not specified, the only Disk Locator
                      assigned to the Device is used.

     One of these resolutions when needed

        storeId                    GUID of the Store that is needed when a
                                   diskLocatorName is used.

        Guid[] StoreId: GUID of the Store that is needed when a
                      DiskLocatorName is used.

        storeName                  Name of the Store that is needed when a
                                   diskLocatorName is used.

        string[] StoreName: Name of the Store that is needed when a
                      DiskLocatorName is used.

     Instead of a parameter that matches one of the members listed

        PvsObject[] Object: PvsObjects with the members below can be used as
                      the Object parameter or from a pileline:

          DeviceId or DiskLocatorId

     Optional

        SwitchParameter Confirm: The impact of this operation is "low". If -
                      Confirm is specified, the operation will be
                      confirmed. $ConfirmPreference can be set to "low" to
                      have confirmation without the Confirm parameter.
```

Mcli-Run StartStreamService

Start the Stream Service on a Server or all Servers in a Site.

Start-PvsStreamService

Start the Stream Service on a Server or all Servers in a Site.

```
     -p                            Parameters needed for this Run.

       One of these required
```

```
        serverId                  GUID of the Server to start the Stream
                                  Service.

    Guid[] Guid or ServerId: GUID of the Server to start the Stream
                      Service.

        serverName                Name of the Server to start the Stream
                                  Service.

    string[] Name or ServerName: Name of the Server to start the Stream
                      Service.

        siteId                    GUID of the Site to start the Stream Service
                                  on all Servers.

    Guid[] SiteId: GUID of the Site to start the Stream Service on all
                      Servers.

        siteName                  Name of the Site to start the Stream Service
                                  on all Servers.

    string[] SiteName: Name of the Site to start the Stream Service on
                      all Servers.

        Instead of a parameter that matches one of the members listed

    PvsObject[] Object: PvsObjects with the members below can be used as
                      the Object parameter or from a pileline:

      ServerId or SiteId

    Optional

    SwitchParameter Confirm: The impact of this operation is "low". If -
                      Confirm is specified, the operation will be
                      confirmed. $ConfirmPreference can be set to "low" to
                      have confirmation without the Confirm parameter.
```

Mcli-Run StopDeviceDiskTempVersionMode

```
    Unset the Temporary Disk Version for the specified Device. The Device must
      not currently be booted. Once removed, the Device again uses any
      currently assigned vDisk when booted.
```

Stop-PvsDeviceDiskTempVersionMode

```
    Unset the Temporary Disk Version for the specified Device. The Device must
      not currently be booted. Once removed, the Device again uses any
      currently assigned vDisk when booted.

      -p                          Parameters needed for this Run.

      One of these required

      deviceId                  GUID of the Device to no longer use a
                                 temporary disk version for.

      Guid[] Guid or DeviceId: GUID of the Device to no longer use a
                        temporary disk version for.

      deviceName                Name of the Device to no longer use a
                                 temporary disk version for.

      string[] Name or DeviceName: Name of the Device to no longer use a
                        temporary disk version for.
```

deviceMac   MAC of the Device to no longer use a
                            temporary disk version for.

  PvsPhysicalAddress[] DeviceMac: MAC of the Device to no longer use a
                  temporary disk version for.

 diskLocatorId   GUID of the DiskLocator to no longer use
                            temporary disk versions for.

  Guid[] DiskLocatorId: GUID of the DiskLocator to no longer use
                  temporary disk versions for.

 siteId   GUID of the Site for resolution of the
                            diskLocatorName.

  Guid[] SiteId: GUID of the Site for resolution of the
                  DiskLocatorName.

 siteName   Name of the Site for resolution of the
                            diskLocatorName.

  string[] SiteName: Name of the Site for resolution of the
                  DiskLocatorName.

or this required & resolution

 diskLocatorName   Name of the DiskLocator to no longer use
                            temporary disk versions for.

  string[] DiskLocatorName: Name of the DiskLocator to no longer use
                  temporary disk versions for.

This optional & resolution

 version   Version of the DiskLocator to no longer use
                            temporary disk versions for.

  string[] Version: Version of the DiskLocator to no longer use
                  temporary disk versions for.

One of these resolutions when needed

 siteId   GUID of the Site.

  Guid[] SiteId: GUID of the Site for resolution of the
                  DiskLocatorName.

 siteName   Name of the Site.

  string[] SiteName: Name of the Site for resolution of the
                  DiskLocatorName.

One of these resolutions when needed

 diskLocatorId   GUID of the DiskLocator to no longer use
                            temporary disk versions for.

  Guid[] DiskLocatorId: GUID of the DiskLocator to no longer use
                  temporary disk versions for.

 diskLocatorName   Name of the DiskLocator to no longer use
                            temporary disk versions for.

  string[] DiskLocatorName: Name of the DiskLocator to no longer use
                  temporary disk versions for.

One of these resolutions when needed

```
          storeId                 GUID of the Store that is needed when a
                                  diskLocatorName is used.

        Guid[] StoreId: GUID of the Store that is needed when a
                        DiskLocatorName is used.

          storeName               Name of the Store that is needed when a
                                  diskLocatorName is used.

        string[] StoreName: Name of the Store that is needed when a
                        DiskLocatorName is used.

        Instead of a parameter that matches one of the members listed

        PvsObject[] Object: PvsObjects with the members below can be used as
                        the Object parameter or from a pileline:

          DeviceId, DiskLocatorId or SiteId

        Optional

        SwitchParameter Confirm: The impact of this operation is "low". If -
                        Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "low" to
                        have confirmation without the Confirm parameter.
```

Mcli-Run StopStreamService

```
     Stop the Stream Service on a Server or all Servers in a Site.
```

Stop-PvsStreamService

```
     Stop the Stream Service on a Server or all Servers in a Site.

       -p                          Parameters needed for this Run.

       One of these required

         serverId                GUID of the Server to stop the Stream
                                 Service.

          Guid[] Guid or ServerId: GUID of the Server to stop the Stream
                          Service.

         serverName              Name of the Server to stop the Stream
                                 Service.

         string[] Name or ServerName: Name of the Server to stop the Stream
                         Service.

         siteId                  GUID of the Site to stop the Stream Service
                                 on all Servers.

          Guid[] SiteId: GUID of the Site to stop the Stream Service on all
                          Servers.

         siteName                Name of the Site to stop the Stream Service
                                 on all Servers.

         string[] SiteName: Name of the Site to stop the Stream Service on all
                         Servers.

        Instead of a parameter that matches one of the members listed

         PvsObject[] Object: PvsObjects with the members below can be used as
                         the Object parameter or from a pileline:
```

```
                    ServerId or SiteId

             Optional

               SwitchParameter Confirm: The impact of this operation is "medium". If
                             -Confirm is specified, the operation will be
                             confirmed. $ConfirmPreference can be set to "medium"
                             or "low" to have confirmation without the Confirm
                             parameter.
```

Mcli-Run UnlockAllDisk

```
    Remove all locks for a Disk.
```

Unlock-PvsAllDisk

```
    Remove all locks for a Disk.

      -p                          Parameters needed for this Run.

      This required

        diskLocatorId             GUID of the Disk Locator to remove all locks
                                    for the Disk.

          Guid[] Guid or DiskLocatorId: GUID of the Disk Locator to remove all
                        locks for the Disk.

      or this required & resolution

        diskLocatorName           Name of Disk Locator to remove all locks for
                                    the Disk.

          string[] Name or DiskLocatorName: Name of Disk Locator to remove all
                        locks for the Disk.

      One of these resolutions when needed

        siteId                    GUID of the Site.

          Guid[] SiteId: GUID of the Site.

        siteName                  Name of the Site.

          string[] SiteName: Name of the Site.

      One of these resolutions when needed

        storeId                   GUID of the Store that is needed when a
                                    diskLocatorName is used.

          Guid[] StoreId: GUID of the Store that is needed when a
                        DiskLocatorName is used.

        storeName                 Name of the Store that is needed when a
                                    diskLocatorName is used.

          string[] StoreName: Name of the Store that is needed when a
                        DiskLocatorName is used.

      Instead of a parameter that matches one of the members listed

          PvsObject[] Object: PvsObjects with the members below can be used as
                        the Object parameter or from a pileline:

            DiskLocatorId

      Optional
```

Mcli-Run UnlockDisk

Remove lock for the Disk.

Unlock-PvsDisk

Remove lock for the Disk.

```
-p                         Parameters needed for this Run.
 This required
  diskLocatorId            GUID of the Disk Locator to remove a lock
                            for the Disk.

    Guid[] Guid or DiskLocatorId: GUID of the Disk Locator to remove a
                    lock for the Disk.

 or this required & resolution
  diskLocatorName          Name of the Disk Locator to remove a lock
                            for the Disk.

    string[] Name or DiskLocatorName: Name of the Disk Locator to remove
                    a lock for the Disk.

 Optional
  ownerId                  GUID of the Owner of the Disk Lock.

    Guid[] OwnerId: GUID of the Owner of the Disk Lock.

 One of these resolutions when needed
  siteId                   GUID of the Site.

    Guid[] SiteId: GUID of the Site.

  siteName                 Name of the Site.

    string[] SiteName: Name of the Site.

 One of these resolutions when needed
  storeId                  GUID of the Store that is needed when a
                            diskLocatorName is used.

    Guid[] StoreId: GUID of the Store that is needed when a
                    DiskLocatorName is used.

  storeName                Name of the Store that is needed when a
                            diskLocatorName is used.

    string[] StoreName: Name of the Store that is needed when a
                    DiskLocatorName is used.

 Instead of a parameter that matches one of the members listed

    PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

      DiskLocatorId
```

Mcli-Run UnmapDisk

No longer Map the Disk.

Dismount-PvsDisk

No longer Map the Disk.

Mcli-Run ValidateDirectory

Validate a Directory on the Server specified.

Test-PvsDirectory

Validate a Directory on the Server specified.

-p   Parameters needed for this Run.

One of these required

  serverId   GUID of the Server to validate a Directory on.

    Guid[] Guid or ServerId: GUID of the Server to validate a Directory on.

  serverName   Name of the Server to validate a Directory on.

    string[] Name or ServerName: Name of the Server to validate a Directory on.

This required

  path   Path of the Directory to validate.

    string[] Path: Path of the Directory to validate.

This optional

  readOnly   1 when the directory should be validated for a managed read-only store.

    SwitchParameter ReadOnly: If -ReadOnly is specified, the directory should be validated for a managed read-only store.

Instead of a parameter that matches one of the members listed

    PvsObject[] Object: PvsObjects with the members below can be used as the Object parameter or from a pileline:

      ServerId

Optional

    SwitchParameter Confirm: The impact of this operation is "low". If -Confirm is specified, the operation will be confirmed. $ConfirmPreference can be set to "low" to have confirmation without the Confirm parameter.

Mcli-RunWithReturn

```
Mcli-RunWithReturn commandName[ -p name=value[, name2=value2]]
```

Mcli-RunWithReturn Boot

Boot a Device, Collection or View. Returns a taskId of the Task being run.
With the taskId, call ReturnWithReturn TaskStatus to get the percent
complete, Get Task to get the results, and Run CancelTask to stop it
early. The Task result has the deviceName/value of the devices that
succeeded in the first name/value pair of each record of the XML.

Start-PvsDeviceBoot

Boot a Device, Collection or View. Returns a PvsTask of the Task being run.
With the PvsTask, call Get-PvsTaskStatus to get the percent complete,
Get-PvsTask to get the results, and Stop-PvsTask to stop it early. The
PvsTask Results has the DeviceName/value of the devices that succeeded in
the first name/value pair of each record of the XML.

```
-p                          Parameters needed for this Run.

 One of these required

  deviceId                  GUID of the Device to Boot.

    Guid[] Guid or DeviceId: GUID of the Device to Boot.

  deviceName                Name of the Device to Boot.

    string[] Name or DeviceName: Name of the Device to Boot.

  deviceMac                 MAC of the Device to Boot.

    PvsPhysicalAddress[] DeviceMac: MAC of the Device to Boot.

  collectionId              GUID of the Collection to Boot all Devices.

    Guid CollectionId: GUID of the Collection to Boot all Devices.

  siteViewId                GUID of the Site View to Boot all Devices.

    Guid SiteViewId: GUID of the Site View to Boot all Devices.

  farmViewId                GUID of the Farm View to Boot all Devices.

    Guid FarmViewId: GUID of the Farm View to Boot all Devices.

  farmViewName              Name of the Farm View to Boot all Devices.

    string FarmViewName: Name of the Farm View to Boot all Devices.

  deviceIdList              Comma delimited list of GUIDs of the Devices
                             to Boot. Maximum length is 8000 characters.

  deviceNameList            Comma delimited list of Names of the Devices
                             to Boot. Maximum length is 8000 characters.

  deviceMacList             Comma delimited list of MACs of the Devices
                             to Boot. Maximum length is 8000 characters.

 or one of these required & resolutions

  collectionName            Name of the Collection to Boot all Devices.

    string CollectionName: Name of the Collection to Boot all Devices.

  siteViewName              Name of the Site View to Boot all Devices.
```

```
            string SiteViewName: Name of the Site View to Boot all Devices.
        One of these resolutions when needed
         siteId                        GUID of the Site.
           Guid SiteId: GUID of the Site.
         siteName                      Name of the Site.
           string SiteName: Name of the Site.
         Instead of a parameter that matches one of the members listed
           PvsObject[] Object: PvsObjects with the members below can be used as
                        the Object parameter or from a pileline:
            DeviceName, CollectionId, SiteViewId or FarmViewId
         Optional
           SwitchParameter Confirm: The impact of this operation is "low". If -
                        Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "low" to
                        have confirmation without the Confirm parameter.
       rv                             If successful, this rv is filled with a
                                      string.
        PvsTask: If successful, the PvsTask object for the process started is
                    returned.
```

Mcli-RunWithReturn CreateDisk

```
    Create a Disk and the Disk Locator for it. Return the GUID of the
    DiskLocator created. If "" is returned as the GUID, then call
    CreateDiskStatus to get the final GUID when processing finishes.
```

Start-PvsCreateDisk

```
    Create a Disk and the Disk Locator for it. Returns the PvsDiskLocator
    created if finished. If not returned, then call CreateDiskStatus to get
    the PvsDiskLocator when processing finishes.
         -p                           Parameters needed for this Run.
        This required
         name                         Name of the Disk file to be created.
           string Name: Name of the Disk file to be created.
        This required
         size                         Size of the disk in Megabytes for a fixed
                                       size disk. Maximum size in Megabytes for a
                                       dynamically sized disk. Min=1, VHD
                                       Max=2088960, VHDX Max=67108864
           UInt64 Size: Size of the disk in Megabytes for a fixed size disk.
                        Maximum size in Megabytes for a dynamically sized
                        disk. Min=1, VHD Max=2088960, VHDX Max=67108864
        One of these required
         storeId                      GUID of the Store that the Disk will be a
                                       member of.
```

Guid StoreId: GUID of the Store that the Disk will be a member of.

 storeName   Name of the Store that the Disk will be a
                              member of.

     string StoreName: Name of the Store that the Disk will be a member
                  of.

One of these required

 siteId   GUID of the Site.

     Guid SiteId: GUID of the Site.

 siteName   Name of the Site.

     string SiteName: Name of the Site.

One of these optional

 serverId   GUID of the only Server for this Disk.

     Guid ServerId: GUID of the only Server for this Disk.

 serverName   Name of the only Server for this Disk.

     string ServerName: Name of the only Server for this Disk.

Optional

 description   Description of the Disk that will be placed
                              in the Disk Locator.

     string Description: Description of the Disk that will be placed in
                  the Disk Locator.

 enabled   The Disk will be created disabled if set to
                              0. It is created Enabled by default.
                              Default=1

     SwitchParameter CreateDiskDisabled: If -CreateDiskDisabled is
                  specified, the Disk will be created disabled. It is
                  created Enabled by default.

 format   Format of the image. Values are: 0 (VHD), 1
                              (VHDX), Default=0

     SwitchParameter VHDX: If -VHDX is specified, VHDX will be used for
                  the format of the image. VHDX has a Block size of 32
                  MB. VHD is the default.

 type   Type that will be created. 1 for dynamic, or
                              0 for fixed. Default=1 for VHDX, 0 for VHD

     SwitchParameter Dynamic: If -Dynamic is specified, a dynamic VHD that
                  will be created. The default is fixed.

 vhdBlockSize   Block size in KB. For VHD it is only used
                              with Dynamic type. Tested sizes for VHD are
                              512, 2048, and 16384. VHD Min=512,
                              Max=16384, Default=2048. For VHDX it is used
                              for all types. Tested size for VHDX is
                              32768. VHDX Min=1024, Max= 262144,
                              Default=32768.

     uint VhdBlockSize: Block size in KB. For VHD it is only used with
                  Dynamic type. Tested sizes for VHD are 512, 2048,
                  and 16384. VHD Min=512, Max=16384, Default=2048. For

logicalSectorSize   Only used with VHDX format. Logical Sector
                            Size. Values are: 512, 4096, Default=512

uint LogicalSectorSize: Only used with VHDX format. Logical Sector
                        Size. Values are: 512, 4096, Default=512

Instead of a parameter that matches one of the members listed

PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

StoreId, SiteId or ServerId

Optional

SwitchParameter Confirm: The impact of this operation is "low". If -
                Confirm is specified, the operation will be
                confirmed. $ConfirmPreference can be set to "low" to
                have confirmation without the Confirm parameter.

rv   If successful, this rv is filled with a
                            string.

PvsDiskLocator: If the create is already finished and successful, the
                PvsDiskLocator is returned.

Mcli-RunWithReturn CreateDiskStatus

Get the Percent Finished for an active CreateDisk. When finished, the GUID
of the DiskLocator created is returned.

Get-PvsCreateDiskStatus

Get the Percent Finished for an active CreateDisk. When finished, the
PvsDiskLocator created is returned.

-p   Parameters needed for this Run.

This required

name   Name of the Disk file that is being created.

string Name: Name of the Disk file that is being created.

One of these required

storeId   GUID of the Store that the Disk will be a
                            member of.

Guid StoreId: GUID of the Store that the Disk will be a member of.

storeName   Name of the Store that the Disk will be a
                            member of.

string StoreName: Name of the Store that the Disk will be a member
                  of.

Instead of a parameter that matches one of the members listed

PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

StoreId

rv   If successful, this rv is filled with a
                                                string.

        UInt32 or PvsDiskLocator: If not finished, the percent complete is
                        returned in an UInt32. If finished and successful, the
                        PvsDiskLocator is returned.

Mcli-RunWithReturn CreateMaintenanceVersion

Create a Maintenance version for the Disk Locator. Return the name of the
new version file.

New-PvsDiskMaintenanceVersion

Create a Maintenance version for the Disk Locator. Returns a PvsDiskVersion
when successful.

        -p   Parameters needed for this Run.

    This required

      diskLocatorId   GUID of the Disk Locator File to Enable Disk
                                  Maintenance on.

        Guid DiskLocatorId: GUID of the Disk Locator File to Enable Disk
                        Maintenance on.

    or this required & resolution

      diskLocatorName   Name of the Disk Locator File to Enable Disk
                                  Maintenance on.

        string DiskLocatorName: Name of the Disk Locator File to Enable Disk
                        Maintenance on.

    One of these resolutions when needed

      siteId   GUID of the Site.

        Guid SiteId: GUID of the Site.

      siteName   Name of the Site.

        string SiteName: Name of the Site.

    One of these resolutions when needed

      storeId   GUID of the Store that is needed when a
                                  diskLocatorName is used.

        Guid StoreId: GUID of the Store that is needed when a DiskLocatorName
                        is used.

      storeName   Name of the Store that is needed when a
                                  diskLocatorName is used.

        string StoreName: Name of the Store that is needed when a
                        DiskLocatorName is used.

    Instead of a parameter that matches one of the members listed

        PvsObject[] Object: PvsObjects with the members below can be used as
                        the Object parameter or from a pileline:

          DiskLocatorId

        Optional

    rv   If successful, this rv is filled with a
                                    string.

Mcli-RunWithReturn DeviceCount

Get count of Devices in a Collection or View.

Get-PvsDeviceCount

Get count of Devices in a Collection or View.

   -p   Parameters needed for this Run.

  One of these required

   collectionId   GUID of the Collection to get the Device
                                    Count of.

     Guid CollectionId: GUID of the Collection to get the Device Count of.

   siteViewId   GUID of the Site View to get the Device
                                    Count of.

     Guid SiteViewId: GUID of the Site View to get the Device Count of.

   farmViewId   GUID of the Farm View to get the Device
                                    Count of.

     Guid FarmViewId: GUID of the Farm View to get the Device Count of.

   farmViewName   Name of the Farm View to get the Device
                                    Count of.

     string FarmViewName: Name of the Farm View to get the Device Count
                    of.

  or one of these required & resolutions

   collectionName   Name of the Collection to get the Device
                                    Count of.

     string CollectionName: Name of the Collection to get the Device Count
                    of.

   siteViewName   Name of the Site View to get the Device
                                    Count of.

     string SiteViewName: Name of the Site View to get the Device Count
                    of.

  One of these resolutions when needed

   siteId   GUID of the Site.

     Guid SiteId: GUID of the Site.

   siteName   Name of the Site.

     string SiteName: Name of the Site.

Instead of a parameter that matches one of the members listed

                       PvsObject[] Object: PvsObjects with the members below can be used as
                                   the Object parameter or from a pileline:

                          CollectionId, SiteViewId or FarmViewId

        rv   If successful, this rv is filled with a
                                              string.

           UInt32: If successful, the numeric value is returned

Mcli-RunWithReturn DiskLocatorCount

      Get count of Disk Locators for a Site and Type.

Get-PvsDiskLocatorCount

    Get count of Disk Locators for a Site and Type.

      -p   Parameters needed for this Run.

        One of these required

         siteId   GUID of the Site to get the Disk Locator
                                          Count of.

            Guid SiteId: GUID of the Site to get the Disk Locator Count of.

         siteName   Name of the Site to get the Disk Locator
                                          Count of.

            string SiteName: Name of the Site to get the Disk Locator Count of.

        One of these optional

         single   Include single server connection when set to
                                          1. If this and All are not included, both
                                          connection types are included.

            SwitchParameter Single: If -Single is specified, include single
                            server connection. If this and All are not included,
                            both connection types are included.

         all   Include all server connections for the store
                                          when set to 1. If this and Single are not
                                          included, both connection types are
                                          included.

            SwitchParameter All: If -All is specified, include all server
                            connections for the store. If this and Single are
                            not included, both connection types are included.

        Instead of a parameter that matches one of the members listed

           PvsObject[] Object: PvsObjects with the members below can be used as
                       the Object parameter or from a pileline:

              SiteId

      rv   If successful, this rv is filled with a
                                            string.

         UInt32: If successful, the numeric value is returned

Mcli-RunWithReturn DisplayMessage

Display a message on a Device, a Site, Collection or View. Returns a taskId of the Task being run. With the taskId, call ReturnWithReturn TaskStatus to get the percent complete, Get Task to get the results, and Run CancelTask to stop it early. The Task result has the deviceName/value of the devices that succeeded in the first name/value pair of each record of the XML.

Start-PvsDisplayMessage

Display a message on a Device, Collection or View. Returns a PvsTask of the Task being run. With the PvsTask, call Get-PvsTaskStatus to get the percent complete, Get-PvsTask to get the results, and Stop-PvsTask to stop it early. The PvsTask Results has the DeviceName/value of the devices that succeeded in the first name/value pair of each record of the XML.

```
-p                          Parameters needed for this Run.

 This required

  message                   Message to display on the Device(s).

    string Message: Message to display on the Device(s).

 One of these required

  deviceId                  GUID of the Device to Display a Message.

    Guid[] DeviceId: GUID of the Device to Display a Message.

  deviceName                Name of the Device to Display a Message.

    string[] DeviceName: Name of the Device to Display a Message.

  deviceMac                 MAC of the Device to Display a Message.

    PvsPhysicalAddress[] DeviceMac: MAC of the Device to Display a
                          Message.

  collectionId              GUID of the Collection to Display a Message
                           on all Devices.

    Guid CollectionId: GUID of the Collection to Display a Message on all
                          Devices.

  siteViewId                GUID of the Site View to Display a Message
                           all Devices.

    Guid SiteViewId: GUID of the Site View to Display a Message all
                          Devices.

  farmViewId                GUID of the Farm View to Display a Message
                           on all Devices.

    Guid FarmViewId: GUID of the Farm View to Display a Message on all
                          Devices.

  farmViewName              Name of the Farm View to Display a Message
                           on all Devices.

    string FarmViewName: Name of the Farm View to Display a Message on
                          all Devices.
```

deviceIdList	Comma delimited list of GUIDs of the Devices to Display a Message. Maximum length is 8000 characters.
deviceNameList	Comma delimited list of Names of the Devices to Display a Message. Maximum length is 8000 characters.
deviceMacList	Comma delimited list of MACs of the Devices to Display a Message. Maximum length is 8000 characters.

or one of these required & resolutions

collectionName	Name of the Collection to Display a Message on all Devices.

 string CollectionName: Name of the Collection to Display a Message on all Devices.

siteViewName	Name of the Site View to Display a Message on all Devices.

 string SiteViewName: Name of the Site View to Display a Message on all Devices.

One of these resolutions when needed

siteId	GUID of the Site.

 Guid SiteId: GUID of the Site.

siteName	Name of the Site.

 string SiteName: Name of the Site.

Instead of a parameter that matches one of the members listed

 PvsObject[] Object: PvsObjects with the members below can be used as the Object parameter or from a pileline:

  DeviceName, CollectionId, SiteViewId or FarmViewId

 Optional

 SwitchParameter Confirm: The impact of this operation is "low". If -Confirm is specified, the operation will be confirmed. $ConfirmPreference can be set to "low" to have confirmation without the Confirm parameter.

rv	If successful, this rv is filled with a string.

 PvsTask: If successful, the PvsTask object for the process started is returned.

Mcli-RunWithReturn Exists

Return 1 if a Site, Server, Collection, View, Device, Store, Update Task or Virtual Hosting Pool Name is already used. If a collectionName, siteViewName, updateTaskName or virtualHostingPoolName is specified, siteName or siteId must be included.

Get-PvsExists

Return true if a Site, Server, Collection, View, Device, Store, Update Task
   or Virtual Hosting Pool Name is already used. If a CollectionName,
   SiteViewName, UpdateTaskName or VirtualHostingPoolName is specified,
   SiteName or SiteId must be included.

   -p   Parameters needed for this Run.
   One of these required

   siteName   Name of the Site.

      string SiteName: Name of the Site.

   serverName   Server name, to see if it is already used in
                               the Farm.

      string ServerName: Server name, to see if it is already used in the
                   Farm.

   farmViewName   Farm View name, to see if it is already used
                               in the Farm.

      string FarmViewName: Farm View name, to see if it is already used in
                   the Farm.

   deviceName   Device name, to see if it is already used in
                               the Farm.

      string DeviceName: Device name, to see if it is already used in the
                   Farm.

   deviceMac   Device MAC, to see if it is already used in
                               the Farm.

      PvsPhysicalAddress[] DeviceMac: Device MAC, to see if it is already
                   used in the Farm.

   storeName   Store name, to see if it is already used.

      string StoreName: Store name, to see if it is already used.

   or one of these required & resolutions

   collectionName   Collection name, to see if it is already
                               used in a Site.

      string CollectionName: Collection name, to see if it is already used
                   in a Site.

   siteViewName   Site View name, to see if it is already used
                               in the Site.

      string SiteViewName: Site View name, to see if it is already used in
                   the Site.

   virtualHostingPoolName   Virtual Hosting Pool name, to see if it is
                               already used in a Site.

      string VirtualHostingPoolName: Virtual Hosting Pool name, to see if
                   it is already used in a Site.

   updateTaskName   Update Task name, to see if it is already
                               used in a Site.

      string UpdateTaskName: Update Task name, to see if it is already used
                   in a Site.

```
        diskLocatorName              DiskLocator name, to see if it is already
                                      used in a Site.

        string DiskLocatorName: DiskLocator name, to see if it is already
                      used in a Site.

    One of these resolutions when needed

    siteName                         Name of the Site.

        string SiteName: Name of the Site.

    siteId                           GUID of the Site.

        Guid SiteId: GUID of the Site.

    One of these resolutions when needed

    storeId                          GUID of the Store that is needed when a
                                      diskLocatorName is used.

        Guid StoreId: GUID of the Store that is needed when a DiskLocatorName
                      is used.

    storeName                        Name of the Store that is needed when a
                                      diskLocatorName is used.

        string StoreName: Store name, to see if it is already used.

  rv                                 If successful, this rv is filled with a
                                      string.

    Boolean: If successful, $true or $false is returned.
```

Mcli-RunWithReturn ImportDisk

```
Import a Disk. It will add a Disk Locator for the Disk to the Site. A
  manifest file must exist in the Store. The return value is the GUID of
  the DiskLocator added.
```

Import-PvsDisk

```
Import a Disk. It will add a Disk Locator for the Disk to the Site. A
  manifest file must exist in the Store. If successful, the new
  PvsDiskLocator is returned.

  -p                               Parameters needed for this Run.

   This required & resolution

    diskLocatorName              Name of the Disk Locator File. It is unique
                                  within the Store. ASCII Max Length=52

        string Name or DiskLocatorName: Name of the Disk Locator File. It is
                      unique within the Store. ASCII Max Length=52

   One of these optional

    serverId                     GUID of the single Server that this Disk
                                  Locator is assigned to. It is not used with
                                  serverName. Default=null Length=36

        Guid ServerId: GUID of the single Server that this Disk Locator is
                      assigned to. It is not used with ServerName.
                      Default=00000000-0000-0000-0000-000000000000
```

serverName   Name of the single Server that this Disk Locator is assigned to. It is not used with serverId. Default=null

string ServerName: Name of the single Server that this Disk Locator is assigned to. It is not used with ServerId. Default=""

Optional

description   User description. Default="" Max Length=250

string Description: User description. Default="" Max Length=250

menuText   Text that is displayed in the Boot Menu. If this field has no value, the name value is used. Default="" ASCII Max Length=64

string MenuText: Text that is displayed in the Boot Menu. If this field has no value, the name value is used. Default="" ASCII Max Length=64

enabled   1 when this disk can be booted, 0 otherwise. Default=1

SwitchParameter Enabled: True when this disk can be booted, false otherwise. Default=true

rebalanceEnabled   1 when this Server can automatically rebalance Devices, 0 otherwise. Default=0

SwitchParameter RebalanceEnabled: True when this Server can automatically rebalance Devices, false otherwise. Default=false

rebalanceTriggerPercent   Percent over fair load that triggers a dynamic Device rebalance. Min=5, Max=5000, Default=25

uint RebalanceTriggerPercent: Percent over fair load that triggers a dynamic Device rebalance. Min=5, Max=5000, Default=25

subnetAffinity   Qualifier for subnet affinity when assigning a Server. 0=None, 1=Best Effort, 2=Fixed. Min=0, Max=2, Default=0

uint SubnetAffinity: Qualifier for subnet affinity when assigning a Server. 0=None, 1=Best Effort, 2=Fixed. Min=0, Max=2, Default=0

format   Format of the image. Values are: 0 (VHD), 1 (VHDX), Default=0

SwitchParameter VHDX: If -VHDX is specified, VHDX will be used for the format of the image. VHDX has a Block size of 32 MB. VHD is the default.

One of these resolutions when needed

siteId   GUID of the Site.

Guid SiteId: GUID of the Site.

siteName   Name of the Site.

string SiteName: Name of the Site.

One of these resolutions when needed

storeId   GUID of the Store that is needed when a
                                  diskLocatorName is used.

  Guid StoreId: GUID of the Store that is needed when a DiskLocatorName
                  is used.

storeName   Name of the Store that is needed when a
                                  diskLocatorName is used.

  string StoreName: Name of the Store that is needed when a
                  DiskLocatorName is used.

Instead of a parameter that matches one of the members listed

  PvsObject[] Object: PvsObjects with the members below can be used as
                  the Object parameter or from a pileline:

    ServerId

  Optional

  SwitchParameter Confirm: The impact of this operation is "low". If -
                  Confirm is specified, the operation will be
                  confirmed. $ConfirmPreference can be set to "low" to
                  have confirmation without the Confirm parameter.

rv   If successful, this rv is filled with a
                                  string.

  PvsDiskLocator: If successful, the new PvsDiskLocator object is
                  returned.

Mcli-RunWithReturn ListDirectories

Look for Directories or Drives on the Server specified. Return a comma
  delimited list of the Directories or Drives found.

Get-PvsDirectory

Look for Directories or Drives on the Server specified. Return a String
  array of the Directories or Drives found.

  -p   Parameters needed for this Run.

  One of these required

  serverId   GUID of the Server to get a list of
                                  Directories or Drives.

    Guid Guid or ServerId: GUID of the Server to get a list of
                  Directories or Drives.

  serverName   Name of the Server to get a list of
                                  Directories or Drives.

    string Name or ServerName: Name of the Server to get a list of
                  Directories or Drives.

  Optional

  path   Path to get list of Directories for. If not
                                  specified, the Drives are returned.

    string Path: Path to get list of Directories for. If not specified,
                  the Drives are returned.

Instead of a parameter that matches one of the members listed

PvsObject[] Object: PvsObjects with the members below can be used as the Object parameter or from a pileline:

ServerId

rv   If successful, this rv is filled with a string.

string[]: If successful, the array of directory names is returned.

Mcli-RunWithReturn MaintenanceVersionExists

Return 1 if the if the DiskLocator has a maintenance version, 0 otherwise.

Get-PvsMaintenanceVersionExists

Return true if the if the DiskLocator has a maintenance version, false otherwise.

-p   Parameters needed for this Run.

This required

diskLocatorId   DiskLocator GUID, to see if it has a maintenance version.

Guid Guid or DiskLocatorId: DiskLocator GUID, to see if it has a maintenance version.

or this required & resolution

diskLocatorName   DiskLocator name, to see if it has a maintenance version.

string Name or DiskLocatorName: DiskLocator name, to see if it has a maintenance version.

One of these resolutions when needed

siteId   GUID of the Site.

Guid SiteId: GUID of the Site.

siteName   Name of the Site.

string SiteName: Name of the Site.

One of these resolutions when needed

storeId   GUID of the Store that is needed when a diskLocatorName is used.

Guid StoreId: GUID of the Store that is needed when a DiskLocatorName is used.

storeName   Name of the Store that is needed when a diskLocatorName is used.

string StoreName: Name of the Store that is needed when a DiskLocatorName is used.

Instead of a parameter that matches one of the members listed

PvsObject[] Object: PvsObjects with the members below can be used as the Object parameter or from a pileline:

DiskLocatorId

rv   If successful, this rv is filled with a
                                    string.

      Boolean: If successful, $true or $false is returned.

Mcli-RunWithReturn MapDisk

   Map a disk. If successful, the drive letter or an empty string is returned.
      An empty string can be returned if a drive letter was not assigned by the
      operating system before the maxDiskLetterWaitSeconds is used up.

Mount-PvsDisk

   Mount a disk. If successful, the drive letter or an empty string is
      returned. An empty string can be returned if a drive letter was not
      assigned by the operating system before the maxDiskLetterWaitSeconds is
      used up.

      -p   Parameters needed for this Run.

    This required

      diskLocatorId   GUID of the Disk Locator to Map the Disk.

        Guid Guid or DiskLocatorId: GUID of the Disk Locator to Mount the
                        Disk.

     or this required & resolution

      diskLocatorName   Name of the Disk Locator to Map the Disk.

        string Name or DiskLocatorName: Name of the Disk Locator to Mount the
                        Disk.

    One of these optional

      serverId   Specific Server GUID to use to Map the Disk.

        Guid ServerId: Specific Server GUID to use to Mount the Disk.

      serverName   Specific Server Name to use to Map the Disk.

        string ServerName: Specific Server Name to use to Mount the Disk.

    Optional

      maxDiskLetterWaitSeconds Once mapping a disk is successful, this is
                                the maximum amount of seconds spent waiting
                                for the operating system to return a drive
                                letter. If the operating system does not
                                return a drive letter before the maximum
                                wait time, then an empty string is returned.
                                Default=30

        uint MaxDiskLetterWaitSeconds: Once mapping a disk is successful,
                        this is the maximum amount of seconds spent waiting
                        for the operating system to return a drive letter.
                        If the operating system does not return a drive
                        letter before the maximum wait time, then an empty
                        string is returned. Default=30

    One of these resolutions when needed

      siteId   GUID of the Site.

        Guid SiteId: GUID of the Site.

```
         siteName                     Name of the Site.

          string SiteName: Name of the Site.
       One of these resolutions when needed

         storeId                      GUID of the Store that is needed when a
                                      diskLocatorName is used.

          Guid StoreId: GUID of the Store that is needed when a DiskLocatorName
                         is used.

         storeName                    Name of the Store that is needed when a
                                      diskLocatorName is used.

          string StoreName: Name of the Store that is needed when a
                         DiskLocatorName is used.
       Instead of a parameter that matches one of the members listed

          PvsObject[] Object: PvsObjects with the members below can be used as
                         the Object parameter or from a pileline:

            DiskLocatorId or ServerId

          Optional

          SwitchParameter Confirm: The impact of this operation is "low". If -
                         Confirm is specified, the operation will be
                         confirmed. $ConfirmPreference can be set to "low" to
                         have confirmation without the Confirm parameter.

      rv                             If successful, this rv is filled with a
                                      string.

          String: If successful, the String value is returned.
```

Mcli-RunWithReturn MappedDiskId

```
      If there is currently a mapped disk, return the diskLocatorId of the mapped
         disk.
```

Get-PvsMountedDisk

```
      Get the mounted disk, if there is one.

        -p                              Parameters needed for this Run.
       One of these optional

         serverId                     GUID of the Server.

          Guid ServerId: GUID of the Server.

         serverName                   Name of the Server.

          string ServerName: Name of the Server.
       One of these optional

         storeId                      GUID of the Store.

          Guid StoreId: GUID of the Store.

         storeName                    Name of the Store.

          string StoreName: Name of the Store.
       Instead of a parameter that matches one of the members listed
```

PvsObject[] Object: PvsObjects with the members below can be used as
                      the Object parameter or from a pileline:

         ServerId or StoreId
   rv   If successful, this rv is filled with a
                                     string.

      PvsDiskLocator: If successful, the mapped PvsDiskLocator is returned.

Mcli-RunWithReturn MappedDriveLetter

      If there is currently a Mapped Drive, return the Letter of the Drive.

Get-PvsMountedDriveLetter

      If there is currently a Mounted Drive, return the Letter of the Drive.

   rv   If successful, this rv is filled with a
                                     string.

      String: If successful, the String value is returned.

Mcli-RunWithReturn MinimumLastAutoAddDeviceNumber

      Get the minimum that the Device Number of the last Auto Added Device can
      be.

Get-PvsMinimumLastAutoAddDeviceNumber

      Get the minimum that the Device Number of the last Auto Added Device can
      be.

   -p   Parameters needed for this Run.

    This required

     collectionId   GUID of the Collection to get the Minimum
                                     lastAutoAddDeviceNumber for.

        Guid CollectionId: GUID of the Collection to get the Minimum
                      LastAutoAddDeviceNumber for.

    or this required & resolution

     collectionName   Name of the Collection to get the Minimum
                                     lastAutoAddDeviceNumber for.

        string CollectionName: Name of the Collection to get the Minimum
                      LastAutoAddDeviceNumber for.

    Optional

     autoAddPrefix   The string put before the Device Number for
                                     Auto Add.

        string AutoAddPrefix: The string put before the Device Number for
                      Auto Add.

     autoAddSuffix   The string put after the Device Number for
                                     Auto Add.

        string AutoAddSuffix: The string put after the Device Number for Auto
                      Add.

     autoAddNumberLength   The maximum length of the Device Number for
                                     Auto Add. This length plus the autoAddPrefix

length plus the autoAddSuffix length must be
less than 16.

uint AutoAddNumberLength: The maximum length of the Device Number for
Auto Add. This length plus the AutoAddPrefix length
plus the AutoAddSuffix length must be less than 16.

One of these resolutions when needed

siteId   GUID of the Site.

Guid SiteId: GUID of the Site.

siteName   Name of the Site.

string SiteName: Name of the Site.

Instead of a parameter that matches one of the members listed

PvsObject[] Object: PvsObjects with the members below can be used as
the Object parameter or from a pileline:

CollectionId

rv   If successful, this rv is filled with a
string.

UInt32: If successful, the numeric value is returned

Mcli-RunWithReturn RebalanceDevices

Rebalance Devices for a Server. When successful, returns the number of
Devices affected.

Invoke-PvsRebalanceDevices

Rebalance Devices for a Server. When successful, returns the number of
Devices affected.

-p   Parameters needed for this Run.

One of these required

serverId   GUID of the Server to Rebalance Devices on,
serverName.

Guid Guid or ServerId: GUID of the Server to Rebalance Devices on,
ServerName.

serverName   Name of the Server to Rebalance Devices on,
serverId.

string Name or ServerName: Name of the Server to Rebalance Devices
on, ServerId.

Instead of a parameter that matches one of the members listed

PvsObject[] Object: PvsObjects with the members below can be used as
the Object parameter or from a pileline:

ServerId

Optional

SwitchParameter Confirm: The impact of this operation is "low". If -
Confirm is specified, the operation will be
confirmed. $ConfirmPreference can be set to "low" to
have confirmation without the Confirm parameter.

rv   If successful, this rv is filled with a string.

UInt32: If successful, the numeric value is returned

Mcli-RunWithReturn Reboot

Reboot a Device, Collection or View. Returns a taskId of the Task being run. With the taskId, call ReturnWithReturn TaskStatus to get the percent complete, Get Task to get the results, and Run CancelTask to stop it early. The Task result has the deviceName/value of the devices that succeeded in the first name/value pair of each record of the XML.

Start-PvsDeviceReboot

Reboot a Device, Collection or View. Returns a PvsTask of the Task being run. With the PvsTask, call Get-PvsTaskStatus to get the percent complete, Get-PvsTask to get the results, and Stop-PvsTask to stop it early. The PvsTask Results has the DeviceName/value of the devices that succeeded in the first name/value pair of each record of the XML.

-p   Parameters needed for this Run.

 One of these required

  deviceId   GUID of the Device to Reboot.

   Guid[] DeviceId: GUID of the Device to Reboot.

  deviceName   Name of the Device to Reboot.

   string[] DeviceName: Name of the Device to Reboot.

  deviceMac   MAC of the Device to Reboot.

   PvsPhysicalAddress[] DeviceMac: MAC of the Device to Reboot.

  collectionId   GUID of the Collection to Reboot all Devices.

   Guid CollectionId: GUID of the Collection to Reboot all Devices.

  siteViewId   GUID of the Site View to Reboot all Devices.

   Guid SiteViewId: GUID of the Site View to Reboot all Devices.

  farmViewId   GUID of the Farm View to Reboot all Devices.

   Guid FarmViewId: GUID of the Farm View to Reboot all Devices.

  farmViewName   Name of the Farm View to Reboot all Devices.

   string FarmViewName: Name of the Farm View to Reboot all Devices.

  diskLocatorId   GUID of the DiskLocator to Reboot all Devices.

   Guid DiskLocatorId: GUID of the DiskLocator to Reboot all Devices.

  deviceIdList   Comma delimited list of GUIDs of the Devices to Reboot. Maximum length is 8000 characters.

  deviceNameList   Comma delimited list of Names of the Devices to Reboot. Maximum length is 8000 characters.

```
   deviceMacList              Comma delimited list of MACs of the Devices
                              to Reboot. Maximum length is 8000
                              characters.
or one of these required & resolutions
  collectionName             Name of the Collection to Reboot all
                             Devices.

    string CollectionName: Name of the Collection to Reboot all Devices.

  siteViewName               Name of the Site View to Reboot all Devices.

    string SiteViewName: Name of the Site View to Reboot all Devices.

  diskLocatorName            Name of the DiskLocator to Reboot all
                             Devices.

    string DiskLocatorName: Name of the DiskLocator to Reboot all
                      Devices.
Optional
  message                    Message to display before rebooting the
                             Device(s). Default=The target device is
                             being shutdown remotely by the Console.

    string Message: Message to display before rebooting the Device(s).
                    Default=The target device is being shutdown remotely
                    by the Console.

  delay                      Number of seconds to delay before rebooting
                             the Device(s). Default=10

    uint Delay: Number of seconds to delay before rebooting the
                    Device(s). Default=10

  version                    Version of the Disk to Reboot all Devices
                             for. This is used with diskLocatorId or
                             diskLocatorName.

    uint Version: Version of the Disk to Reboot all Devices for. This is
                    used with DiskLocatorId or DiskLocatorName.
One of these resolutions when needed
  siteId                     GUID of the Site.

    Guid SiteId: GUID of the Site.

  siteName                   Name of the Site.

    string SiteName: Name of the Site.
One of these resolutions when needed
  storeId                    GUID of the Store that is needed when a
                             diskLocatorName is used.

    Guid StoreId: GUID of the Store that is needed when a DiskLocatorName
                    is used.

  storeName                  Name of the Store that is needed when a
                             diskLocatorName is used.

    string StoreName: Name of the Store that is needed when a
                    DiskLocatorName is used.
```

```
            Instead of a parameter that matches one of the members listed

               PvsObject[] Object: PvsObjects with the members below can be used as
                              the Object parameter or from a pileline:

                 DeviceName, CollectionId, SiteViewId, FarmViewId or DiskLocatorId

            Optional

               SwitchParameter Confirm: The impact of this operation is "medium". If
                              -Confirm is specified, the operation will be
                              confirmed. $ConfirmPreference can be set to "medium"
                              or "low" to have confirmation without the Confirm
                              parameter.
      rv                              If successful, this rv is filled with a
                                      string.

         PvsTask: If successful, the PvsTask object for the process started is
                              returned.
```

Mcli-RunWithReturn ReportBug

Report a bug at individual server level or at site level. Return the id of
the long running task.

Start-PvsReportBug

Report a bug at individual server level or at site level. Problem report
can be uploaded to Citrix Systems or can be saved locally in a path
accessible from all servers in the farm.

```
      -p                              Parameters needed for this Run.

       One of these required

        siteId                        GUID of the Site.

          Guid Guid or SiteId: GUID of the Site.

        serverId                      GUID of the Server.

          Guid ServerId: GUID of the Server.

       or one of these required & resolutions

        siteName                      Name of the Site.

          string Name or SiteName: Name of the Site.

        serverName                    Name of the Server.

          string ServerName: Name of the Server.

       This required

        Summary                       Short summary describing the problem.

          string Summary: Short summary describing the problem.

       This optional

        srNumber                      Service Request number of the reported
                                       problem.

          string SrNumber: Service Request number of the reported problem.

       This optional
```

```
      Description                        Description of the reported problem.

         string Description: Description of the reported problem.

      This optional

      Path                               Path where problem report bundle is saved.

         string Path: Path where problem report bundle is saved.

      This optional

      DateTime                           DateTime around which the reported problem
                                         occured.

         DateTime DateTime: DateTime around which the reported problem
                           occured.

      Instead of a parameter that matches one of the members listed

         PvsObject[] Object: PvsObjects with the members below can be used as
                           the Object parameter or from a pileline:

           SiteId or ServerId

         Optional

         SwitchParameter Confirm: The impact of this operation is "low". If -
                           Confirm is specified, the operation will be
                           confirmed. $ConfirmPreference can be set to "low" to
                           have confirmation without the Confirm parameter.

      rv                                 If successful, this rv is filled with a
                                         string.

         PvsTask: If successful, the PvsTask object for the process started is
                           returned.
```

Mcli-RunWithReturn ServerCount

```
      Get count of Servers in a Site.
```

Get-PvsServerCount

```
      Get count of Servers in a Site.

       -p                               Parameters needed for this Run.

       One of these required

       siteId                           GUID of the Site to get the Server Count of.

          Guid Guid or SiteId: GUID of the Site to get the Server Count of.

       siteName                         Name of the Site to get the Server Count of.

          string Name or SiteName: Name of the Site to get the Server Count of.

       Instead of a parameter that matches one of the members listed

          PvsObject[] Object: PvsObjects with the members below can be used as
                           the Object parameter or from a pileline:

            SiteId

      rv                                 If successful, this rv is filled with a
                                         string.

         UInt32: If successful, the numeric value is returned
```

Mcli-RunWithReturn ServerName

Return the name of the Server the SoapServer is running on.

Get-PvsServerName

Return the name of the Server the SoapServer is running on.

rv   If successful, this rv is filled with a
                                      string.

String: If successful, the String value is returned.

Mcli-RunWithReturn ServerStoreActiveDeviceCount

Get the count of Devices currently connected to any vdisk served from the
Store by the Server.

Get-PvsServerStoreActiveDeviceCount

Get the count of Devices currently connected to any vdisk served from the
Store by the Server.

-p   Parameters needed for this Run.

One of these required

serverId   GUID of the Server.

Guid ServerId: GUID of the Server.

serverName   Name of the Server.

string ServerName: Name of the Server.

One of these required

storeId   GUID of the Store.

Guid StoreId: GUID of the Store.

storeName   Name of the Store.

string StoreName: Name of the Store.

Instead of a parameter that matches one of the members listed

PvsObject[] Object: PvsObjects with the members below can be used as
                    the Object parameter or from a pileline:

ServerId or StoreId

rv   If successful, this rv is filled with a
                                      string.

UInt32: If successful, the numeric value is returned

Mcli-RunWithReturn Shutdown

Shutdown a Device, Collection or View. Returns a taskId of the Task being
run. With the taskId, call ReturnWithReturn TaskStatus to get the percent
complete, Get Task to get the results, and Run CancelTask to stop it
early. The Task result has the deviceName/value of the devices that
succeeded in the first name/value pair of each record of the XML.

Start-PvsDeviceShutdown

Shutdown a Device, Collection or View. Returns a PvsTask of the Task being run. With the PvsTask, call Get-PvsTaskStatus to get the percent complete, Get-PvsTask to get the results, and Stop-PvsTask to stop it early. The PvsTask Results has the DeviceName/value of the devices that succeeded in the first name/value pair of each record of the XML.

```
-p                          Parameters needed for this Run.
 One of these required
  deviceId                  GUID of the Device to Shutdown.
    Guid[] DeviceId: GUID of the Device to Shutdown.
  deviceName                Name of the Device to Shutdown.
    string[] DeviceName: Name of the Device to Shutdown.
  deviceMac                 MAC of the Device to Shutdown.
    PvsPhysicalAddress[] DeviceMac: MAC of the Device to Shutdown.
  collectionId              GUID of the Collection to Shutdown all
                            Devices.
    Guid CollectionId: GUID of the Collection to Shutdown all Devices.
  siteViewId                GUID of the Site View to Shutdown all
                            Devices.
    Guid SiteViewId: GUID of the Site View to Shutdown all Devices.
  farmViewId                GUID of the Farm View to Shutdown all
                            Devices.
    Guid FarmViewId: GUID of the Farm View to Shutdown all Devices.
  farmViewName              Name of the Farm View to Shutdown all
                            Devices.
    string FarmViewName: Name of the Farm View to Shutdown all Devices.
  diskLocatorId             GUID of the DiskLocator to Shutdown all
                            Devices.
    Guid DiskLocatorId: GUID of the DiskLocator to Shutdown all Devices.
  deviceIdList              Comma delimited list of GUIDs of the Devices
                            to Shutdown. Maximum length is 8000
                            characters.
  deviceNameList            Comma delimited list of Names of the Devices
                            to Shutdown. Maximum length is 8000
                            characters.
  deviceMacList             Comma delimited list of MACs of the Devices
                            to Shutdown. Maximum length is 8000
                            characters.
 or one of these required & resolutions
  collectionName            Name of the Collection to Shutdown all
                            Devices.
    string CollectionName: Name of the Collection to Shutdown all
                    Devices.
```

siteViewName   Name of the Site View to Shutdown all
                          Devices.

  string SiteViewName: Name of the Site View to Shutdown all Devices.

diskLocatorName   Name of the DiskLocator to Shutdown all
                          Devices.

  string DiskLocatorName: Name of the DiskLocator to Shutdown all
                  Devices.

Optional

 message   Message to display before shutting down the
                          Device(s). Default=The target device is
                          being shutdown remotely by the Console.

  string Message: Message to display before shutting down the
                  Device(s). Default=The target device is being
                  shutdown remotely by the Console.

 delay   Number of seconds to delay before shutting
                          down the Device(s). Default=10

  uint Delay: Number of seconds to delay before shutting down the
                  Device(s). Default=10

 version   Version of the Disk to Shutdown all Devices
                          for. This is used with diskLocatorId or
                          diskLocatorName.

  uint Version: Version of the Disk to Shutdown all Devices for. This
                  is used with DiskLocatorId or DiskLocatorName.

One of these resolutions when needed

 siteId   GUID of the Site.

  Guid SiteId: GUID of the Site.

 siteName   Name of the Site.

  string SiteName: Name of the Site.

One of these resolutions when needed

 storeId   GUID of the Store that is needed when a
                          diskLocatorName is used.

  Guid StoreId: GUID of the Store that is needed when a DiskLocatorName
                  is used.

 storeName   Name of the Store that is needed when a
                          diskLocatorName is used.

  string StoreName: Name of the Store that is needed when a
                  DiskLocatorName is used.

 Instead of a parameter that matches one of the members listed

  PvsObject[] Object: PvsObjects with the members below can be used as
                  the Object parameter or from a pileline:

   DeviceName, CollectionId, SiteViewId, FarmViewId or DiskLocatorId

 Optional

          rv   If successful, this rv is filled with a
                                          string.

Mcli-RunWithReturn StoreFreeSpace

Get the free megabytes available in the Store.

Get-PvsStoreFreeSpace

   -p   Parameters needed for this Run.

  One of these required & resolutions

   storeId   GUID of the Store.

   storeName   Name of the Store.

  One of these resolutions when needed

   serverId   GUID of the Server to use to determine the
                                      free space in the Store.

   serverName   Name of the Server to use to determine the
                                      free space in the Store.

   siteId   GUID of the Site.

   siteName   Name of the Site.

   rv   If successful, this rv is filled with a
                                      string.

Mcli-RunWithReturn TaskStatus

Get the status of a Task in percent complete.

Get-PvsTaskStatus

   -p   Parameters needed for this Run.

```
        This required

        taskId                        Id of the Task to get the Status of.

          uint TaskId: Id of the Task to get the Status of.

        Instead of a parameter that matches one of the members listed

          PvsObject[] Object: PvsObjects with the members below can be used as
                             the Object parameter or from a pileline:

            TaskId

      rv                              If successful, this rv is filled with a
                                      string.

          UInt32: If successful, the numeric value is returned
```

Mcli-RunWithReturn UpdateBDM

```
    Update the BDM Partition for a Device, Collection or View. Returns a taskId
      of the Task being run. With the taskId, call ReturnWithReturn TaskStatus
      to get the percent complete, Get Task to get the results, and Run
      CancelTask to stop it early. The Task result has the deviceName/value of
      the devices that succeeded in the first name/value pair of each record of
      the XML.
```

Start-PvsDeviceUpdateBdm

```
    Update the BDM partition for a Device, Collection or View. Returns a
      PvsTask of the Task being run. With the PvsTask, call Get-PvsTaskStatus
      to get the percent complete, Get-PvsTask to get the results, and Stop-
      PvsTask to stop it early. The PvsTask Results has the DeviceName/value of
      the devices that succeeded in the first name/value pair of each record of
      the XML.

      -p                              Parameters needed for this Run.

       One of these required

        deviceId                      GUID of the Device to Update.

          Guid[] Guid or DeviceId: GUID of the Device to Update.

        deviceName                    Name of the Device to Update.

          string[] Name or DeviceName: Name of the Device to Update.

        collectionId                  GUID of the Collection to Update all BDM
                                      Devices.

          Guid CollectionId: GUID of the Collection to Update all BDM Devices.

        siteViewId                    GUID of the Site View to Update all BDM
                                      Devices.

          Guid SiteViewId: GUID of the Site View to Update all BDM Devices.

        farmViewId                    GUID of the Farm View to Update all BDM
                                      Devices.

          Guid FarmViewId: GUID of the Farm View to Update all BDM Devices.

        farmViewName                  Name of the Farm View to Update all BDM
                                      Devices.

          string FarmViewName: Name of the Farm View to Update all BDM Devices.
```

```
       deviceIdList              Comma delimited list of GUIDs of the Devices
                                 to Update. Maximum length is 8000
                                 characters.
       deviceNameList            Comma delimited list of names of the Devices
                                 to Update. Maximum length is 8000
                                 characters.
  or one of these required & resolutions
     collectionName              Name of the Collection to Update all BDM
                                 Devices.
```

   string CollectionName: Name of the Collection to Update all BDM
                 Devices.

```
     siteViewName                Name of the Site View to Update all
                                 BDMDevices.
```

   string SiteViewName: Name of the Site View to Update all BDMDevices.

```
  One of these resolutions when needed
     siteId                      GUID of the Site.
```

    Guid SiteId: GUID of the Site.

```
     siteName                    Name of the Site.
```

     string SiteName: Name of the Site.

```
     Instead of a parameter that matches one of the members listed
```

    PvsObject[] Object: PvsObjects with the members below can be used as
                 the Object parameter or from a pileline:

     DeviceName, CollectionId, SiteViewId or FarmViewId

   Optional

    SwitchParameter Confirm: The impact of this operation is "low". If -
                 Confirm is specified, the operation will be
                 confirmed. $ConfirmPreference can be set to "low" to
                 have confirmation without the Confirm parameter.

```
   rv                           If successful, this rv is filled with a
                                string.
```

   PvsTask: If successful, the PvsTask object for the process started is
                 returned.

Mcli-RunWithReturn UploadCeip

```
  Perform a one time upload of CEIP data. Return upload Id if successful.
```

Get-PvsUploadCeip

  Perform a one time upload of CEIP data. Return upload Id if successful.

```
    -p                          Parameters needed for this Run.
   This optional
     oneTimeUpload               1 to perform a one time upload.
```

      string OneTimeUpload: If -OneTimeUpload is specified, perform a one
                 time upload.

Optional

SwitchParameter Confirm: The impact of this operation is "low". If -Confirm is specified, the operation will be confirmed. $ConfirmPreference can be set to "low" to have confirmation without the Confirm parameter.

rv   If successful, this rv is filled with a string.

String: If successful, the String value is returned.

Mcli-Set

Mcli-Set commandName[-p name=value[, name2=value2]][-r field=value[, field2=value2]]

Mcli-Set AuthGroup

Set one or more field values for an AuthGroup.

Set-PvsAuthGroup

Set AuthGroup(s) changed values from PvsAuthGroup object(s), or set one or more field values for a PvsAuthGroup.

-p   Parameters needed to identify the record to be changed.

When AuthGroup is not passed the parameters below are used:

One of these required

authGroupId   GUID of the AuthGroup to Set.

Guid Guid or AuthGroupId: GUID of the AuthGroup to Set.

authGroupName   Name of the AuthGroup to Set.

string Name or AuthGroupName: Name of the AuthGroup to Set.

Required

PvsAuthGroup[] AuthGroup: Array of PvsAuthGroup objects with changed property value(s) to be set. The object(s) can come from a pileline.

Optional

SwitchParameter PassThru: If -PassThru is specified, the resulting PvsAuthGroup object(s) are returned.

SwitchParameter Confirm: The impact of this operation is "low". If -Confirm is specified, the operation will be confirmed. $ConfirmPreference can be set to "low" to have confirmation without the Confirm parameter.

-r   Record containing the new values for fields to be changed.

Fields that can have value set

authGroupName   Name of the Active Directory or Windows Group. Max Length=450

string NewName: Name of the Active Directory or Windows Group. Max
                Length=450

description   User description. Default="" Max Length=250

string Description: User description. Default="" Max Length=250

Mcli-Set CeipData

Set one or more field values for CEIP data.

Set-PvsCeipData

Set Ceip changed values from a PvsCeip object, or set one or more field
  values for a PvsCeip.

-p   Parameters needed to identify the record to
                          be changed.

 When CeipData is not passed the parameters below are used:

 Optional

  uuid   CEIP UUID of this Farm. This is optional
                             since there is only one.

    string Uuid: CEIP UUID of this Farm. This is optional since there is
                 only one.

   Required

    PvsCeipData[] CeipData: Array of PvsCeipData objects with changed
                    property value(s) to be set. The object(s) can come
                    from a pileline.

   Optional

    SwitchParameter PassThru: If -PassThru is specified, the resulting
                    PvsCeipData object(s) are returned.

    SwitchParameter Confirm: The impact of this operation is "low". If -
                    Confirm is specified, the operation will be
                    confirmed. $ConfirmPreference can be set to "low" to
                    have confirmation without the Confirm parameter.

-r   Record containing the new values for fields
                          to be changed.

  Fields that can have value set

  enabled   1 if CEIP is enabled, otherwise 0. Min=0,
                             Max=1

   uint Enabled: 1 if CEIP is enabled, otherwise 0. Min=0, Max=1

  nextUpload   Date and time next CEIP upload is due if
                             enabled is 1. Default=null

   DateTime NextUpload: Date and time next CEIP upload is due if enabled
                 is 1. Default=Empty

  inProgress   1 if an upload is currently in progress,
                             otherwise 0. Default=0

   uint InProgress: 1 if an upload is currently in progress, otherwise
                 0. Default=0

```
         serverId                   ID of server that is currently uploading,
                                     null if inProgress is 0. Default=null
                                     Length=36
```

```
         oneTimeUpload              1 to perform a one time upload. Default=0
```

Mcli-Set CisData

```
    Set one or more field values for CIS data.
```

Set-PvsCisData

Set one or more field values for CIS data.

```
   -p                              Parameters needed to identify the record to
                                   be changed.
```

When CisData is not passed the parameters below are used:

```
   Optional
```

```
    cisDataId                      CIS UUID of this Farm. This is optional
                                   since there is only a single record.
```

      Guid Guid or CisDataId: CIS UUID of this Farm. This is optional since
                   there is only a single record.

      Required

       PvsCisData[] CisData: Array of PvsCisData objects with changed
                   property value(s) to be set. The object(s) can come
                   from a pileline.

      Optional

       SwitchParameter PassThru: If -PassThru is specified, the resulting
                   PvsCisData object(s) are returned.

       SwitchParameter Confirm: The impact of this operation is "low". If -
                   Confirm is specified, the operation will be
                   confirmed. $ConfirmPreference can be set to "low" to
                   have confirmation without the Confirm parameter.

```
    -r                             Record containing the new values for fields
                                   to be changed.
```

   Fields that can have value set

```
     userName                      Username used to obtain the token
                                    Default=null Max Length=255
```

      string UserName: Username used to obtain the token Default="" Max
                   Length=255

```
     path                          Path where the last problem report bundle
                                   was saved Default=null Max Length=255
```

      string Path: Path where the last problem report bundle was saved
                   Default="" Max Length=255

password   Password of the user required to obtain the
                                 token. This is required only by Set and Add

           string Password: Password of the user required to obtain the token.
                          This is required only by Set and Add

Mcli-Set Collection

Set one or more field values for a Collection.

Set-PvsCollection

Set Collection(s) changed values from PvsCollection object(s), or set one
  or more field values for a PvsCollection.

   -p   Parameters needed to identify the record to
                                  be changed.

   When Collection is not passed the parameters below are used:

   This required

    collectionId   GUID of the Collection to Set.

      Guid Guid or CollectionId: GUID of the Collection to Set.

    or this required & resolution

    collectionName   Name of the Collection to Set.

      string Name or CollectionName: Name of the Collection to Set.

   One of these resolutions when needed

    siteId   GUID of the Site.

      Guid SiteId: GUID of the Site.

    siteName   Name of the Site.

      string SiteName: Name of the Site.

    Required

      PvsCollection[] Collection: Array of PvsCollection objects with
                       changed property value(s) to be set. The object(s)
                       can come from a pileline.

    Optional

      SwitchParameter PassThru: If -PassThru is specified, the resulting
                       PvsCollection object(s) are returned.

      SwitchParameter Confirm: The impact of this operation is "low". If -
                       Confirm is specified, the operation will be
                       confirmed. $ConfirmPreference can be set to "low" to
                       have confirmation without the Confirm parameter.

   -r   Record containing the new values for fields
                                  to be changed.

    Fields that can have value set

    collectionName   Name of the Collection. It is unique within
                                  the Site. Max Length=50

      string NewName: Name of the Collection. It is unique within the Site.
                      Max Length=50

description   User description. Default="" Max Length=250

string Description: User description. Default="" Max Length=250

templateDeviceId   GUID of a Device in the Collection whose
                              settings are used for initial values of new
                              Devices. Not used with templateDeviceName.
                              Default=null Length=36

Guid TemplateDeviceId: GUID of a Device in the Collection whose
                       settings are used for initial values of new Devices.
                       Not used with templateDeviceName. Default=00000000-
                       0000-0000-0000-000000000000

templateDeviceName   Name of a Device in the Collection whose
                              settings are used for initial values of new
                              Devices. Not used with templateDeviceId.
                              Default=null

string TemplateDeviceName: Name of a Device in the Collection whose
                           settings are used for initial values of new Devices.
                           Not used with TemplateDeviceId. Default=""

lastAutoAddDeviceNumber   The Device Number of the last Auto Added
                              Device. Default=0

uint LastAutoAddDeviceNumber: The Device Number of the last Auto
                              Added Device. Default=0

enabled   1 when Devices in the Collection can be
                              booted, 0 otherwise. Default=1

bool Enabled: True when Devices in the Collection can be booted,
              false otherwise. Default=true

autoAddPrefix   The string put before the Device Number for
                              Auto Add. Default=null ASCII computer name
                              characters no end digit Max Length=12

string AutoAddPrefix: The string put before the Device Number for
                      Auto Add. Default="" ASCII computer name characters
                      no end digit Max Length=12

autoAddSuffix   The string put after the Device Number for
                              Auto Add. Default=null ASCII computer name
                              characters no begin digit Max Length=12

string AutoAddSuffix: The string put after the Device Number for Auto
                      Add. Default="" ASCII computer name characters no
                      begin digit Max Length=12

autoAddZeroFill   1 when zeros be placed before the Device
                              Number up to the autoAddNumberLength for
                              Auto Add, 0 otherwise. Default=1

bool AutoAddZeroFill: True when zeros be placed before the Device
                      Number up to the AutoAddNumberLength for Auto Add,
                      false otherwise. Default=true

autoAddNumberLength   The maximum length of the Device Number for
                              Auto Add. This length plus the autoAddPrefix
                              length plus the autoAddSuffix length must be
                              less than 16. Required that

```
                                     ((lenautoAddPrefix+lenautoAddSuffix)+autoAdd
                                     NumberLength)<=15. Min=3, Max=9, Default=4
          uint AutoAddNumberLength: The maximum length of the Device Number for
                         Auto Add. This length plus the AutoAddPrefix length
                         plus the AutoAddSuffix length must be less than 16.
                         Required that
                         ((lenautoAddPrefix+lenautoAddSuffix)+AutoAddNumberLe
                         ngth)<=15. Min=3, Max=9, Default=4
```

Mcli-Set Device

```
    Set one or more field values for one or more Devices.
```

Set-PvsDevice

```
    Set Device(s) changed values from PvsDevice object(s), or set one or more
       field values for one or more PvsDevices.

      -p                         Parameters needed to identify the record to
                                 be changed.

       When Device is not passed the parameters below are used:
      One of these required
       deviceId                  GUID of the Device to Set.

         Guid Guid or DeviceId: GUID of the Device to Set.

       deviceName                Name of the Device to Set.

          string Name or DeviceName: Name of the Device to Set.

       deviceMac                 MAC of the Device to Set.

         PvsPhysicalAddress DeviceMac: MAC of the Device to Set.

       collectionId              GUID of the Collection to set all Devices.
                                    deviceName and deviceMac cannot be set.

         Guid CollectionId: GUID of the Collection to set all Devices.
                        DeviceName and DeviceMac cannot be set.

       siteViewId                GUID of the Site View to set all Devices.
                                    deviceName and deviceMac cannot be set.

         Guid SiteViewId: GUID of the Site View to set all Devices. DeviceName
                        and DeviceMac cannot be set.

       farmViewId                GUID of the Farm View to set all Devices.
                                    deviceName and deviceMac cannot be set.

         Guid FarmViewId: GUID of the Farm View to set all Devices. DeviceName
                        and DeviceMac cannot be set.

       farmViewName              Name of the Farm View to set all Devices.
                                    deviceName and deviceMac cannot be set.

          string FarmViewName: Name of the Farm View to set all Devices.
                        DeviceName and DeviceMac cannot be set.

      or one of these required & resolutions

       collectionName            Name of the Collection to set all Devices.
                                    deviceName and deviceMac cannot be set.
```

```
            string CollectionName: Name of the Collection to set all Devices.
                         DeviceName and DeviceMac cannot be set.
      siteViewName                 Name of the Site View to set all Devices.
                                    deviceName and deviceMac cannot be set.

            string SiteViewName: Name of the Site View to set all Devices.
                         DeviceName and DeviceMac cannot be set.

  One of these resolutions when needed
    siteId                         GUID of the Site.

      Guid SiteId: GUID of the Site.

    siteName                       Name of the Site.

      string SiteName: Name of the Site.

    Required

      PvsDevice[] Device: Array of PvsDevice objects with changed property
                   value(s) to be set. The object(s) can come from a
                   pileline.

    Optional

      SwitchParameter PassThru: If -PassThru is specified, the resulting
                   PvsDevice object(s) are returned.

      SwitchParameter Confirm: The impact of this operation is "low". If -
                   Confirm is specified, the operation will be
                   confirmed. $ConfirmPreference can be set to "low" to
                   have confirmation without the Confirm parameter.

  -r                               Record containing the new values for fields
                                    to be changed.

  Fields that can have value set
    deviceName                     Computer name with no spaces. ASCII computer
                                    name characters Max Length=15

      string NewName: Computer name with no spaces. ASCII computer name
                   characters Max Length=15

    description                    User description. Default="" Max Length=250

      string Description: User description. Default="" Max Length=250

    deviceMac                      Ethernet address as a string in the form XX-
                                    XX-XX-XX-XX-XX. Uniquely identifies the
                                    Device. Length=17

      PvsPhysicalAddress DeviceMac: Ethernet address can have the form XX-
                   XX-XX-XX-XX-XX. Uniquely identifies the Device.

    bootFrom                       Device to boot from. Choices are 1 for
                                    vDisk, 2 for Hard Disk, and 3 for Floppy.
                                    This cannot be Set for a Device with
                                    Personal vDisk. Min=1, Max=3, Default=1

      uint BootFrom: Device to boot from. Choices are 1 for vDisk, 2 for
                   Hard Disk, and 3 for Floppy. This cannot be Set for
                   a Device with Personal vDisk. Min=1, Max=3,
                   Default=1
```

| className | Used by Automatic Update feature to match new versions of Disks to a Device. This cannot be Set for a Device with Personal vDisk. Default="" Max Length=41 |

string ClassName: Used by Automatic Update feature to match new versions of Disks to a Device. This cannot be Set for a Device with Personal vDisk. Default="" Max Length=41

| port | UDP port to use with Stream Service. Min=1025, Max=65534, Default=6901 |

uint Port: UDP port to use with Stream Service. Min=1025, Max=65534, Default=6901

| enabled | 1 when it can be booted, 0 otherwise. This cannot be Set for a Device with Personal vDisk. Default=1 |

bool Enabled: True when it can be booted, false otherwise. This cannot be Set for a Device with Personal vDisk. Default=true

| localDiskEnabled | If there is a local disk menu choice for the Device, this is 1. This cannot be Set for a Device with Personal vDisk. Default=0 |

bool LocalDiskEnabled: If there is a local disk menu choice for the Device, this is true. This cannot be Set for a Device with Personal vDisk. Default=false

| authentication | Device log in authentication. Choices are 0 for none, 1 for User Name/Password, and 2 for Extern. This cannot be Set for a Device with Personal vDisk. Min=0, Max=2, Default=0 |

uint Authentication: Device log in authentication. Choices are 0 for none, 1 for User Name/Password, and 2 for Extern. This cannot be Set for a Device with Personal vDisk. Min=0, Max=2, Default=0

| user | Name of user to authenticate before the boot process continues. This cannot be Set for a Device with Personal vDisk. Default="" ASCII Max Length=20 |

string User: Name of user to authenticate before the boot process continues. This cannot be Set for a Device with Personal vDisk. Default="" ASCII Max Length=20

| password | Password of user to authenticate before the boot process continues. This cannot be Set for a Device with Personal vDisk. Default="" ASCII Max Length=100 |

string Password: Password of user to authenticate before the boot process continues. This cannot be Set for a Device with Personal vDisk. Default="" ASCII Max Length=100

| adTimestamp | The time the Active Directory machine account password was generated. Do not set |

this field, it is only set internally by
PVS. Default=0

uint AdTimestamp: The time the Active Directory machine account
password was generated. Do not set this field, it is
only set internally by PVS. Default=0

adSignature   The signature of the Active Directory
                               machine account password. Do not set this
                               field, it is only set internally by PVS.
                               Default=0

uint AdSignature: The signature of the Active Directory machine
account password. Do not set this field, it is only
set internally by PVS. Default=0

adPassword   The Active Directory machine account
                               password. Do not set this field, it is only
                               set internally by PVS. Default="" ASCII Max
                               Length=256

string AdPassword: The Active Directory machine account password. Do
not set this field, it is only set internally by
PVS. Default="" ASCII Max Length=256

logLevel   Level to perform logging at. Values are: 0
                               (None), 1 (Fatal), 2 (Error), 3 (Warning), 4
                               (Info), 5 (Debug), and 6 (Trace). Min=0,
                               Max=6, Default=0

uint LogLevel: Level to perform logging at. Values are: 0 (None), 1
(Fatal), 2 (Error), 3 (Warning), 4 (Info), 5
(Debug), and 6 (Trace). Min=0, Max=6, Default=0

domainName   Fully qualified name of the domain that the
                               Device belongs to. Do not set this field, it
                               is only set internally by PVS. Default=null
                               Max Length=255

string DomainName: Fully qualified name of the domain that the Device
belongs to. Do not set this field, it is only set
internally by PVS. Default="" Max Length=255

domainObjectSID   The value of the objectSID AD attribute of
                               the same name for the Device's computer
                               account. Do not set this field, it is only
                               set internally by PVS. Default=null Max
                               Length=186

string DomainObjectSID: The value of the objectSID AD attribute of
the same name for the Device's computer account. Do
not set this field, it is only set internally by
PVS. Default="" Max Length=186

domainControllerName   The name of the DC used to create the host's
                               computer account. Do not set this field, it
                               is only set internally by PVS. Default=null
                               Max Length=4000

string DomainControllerName: The name of the DC used to create the
host's computer account. Do not set this field, it
is only set internally by PVS. Default="" Max
Length=4000

domainTimeCreated   The time that the computer account was
                         created. Do not set this field, it is only
                         set internally by PVS. Format is: YYYY-MM-DD
                         HH:MM:SS.SSS Default=null

 DateTime DomainTimeCreated: The time that the computer account was
                  created. Has the date and time including
                  milliseconds. Do not set this field, it is only set
                  internally by PVS. Default=Empty

type   1 when it performs test of Disks, 2 when it
                         performs maintenance on Disks, 3 when it has
                         a Personal vDisk, 4 when it has a Personal
                         vDisk and performs tests, 0 otherwise. A
                         Device with type 0 - 3 can only be Set to 0
                         - 3, and a Device with type 3 - 4 can only
                         be Set to 3 - 4. Min=0, Max=4, Default=0

  uint Type: 1 when it performs test of Disks, 2 when it performs
                  maintenance on Disks, 3 when it has a Personal
                  vDisk, 4 when it has a Personal vDisk and performs
                  tests, 0 otherwise. A Device with type 0 - 3 can
                  only be Set to 0 - 3, and a Device with type 3 - 4
                  can only be Set to 3 - 4. Min=0, Max=4, Default=0

localWriteCacheDiskSize The size in GB to format the Device cache
                         file disk. If the value is 0, then the disk
                         is not formatted. Min=0, Max=2048, Default=0

  uint LocalWriteCacheDiskSize: The size in GB to format the Device
                  cache file disk. If the value is 0, then the disk is
                  not formatted. Min=0, Max=2048, Default=0

bdmBoot   Use PXE boot when set to 0, BDM boot when
                         set to 1. Default is PXE Default=0

  bool BdmBoot: Use PXE boot when set to false, BDM boot when set to
                  true. Default is PXE Default=false

bdmType   Use PXE boot when set to 0, BDM (Bios) boot
                         when set to 1 and BDM (Uefi) boot when set
                         to 2. Default=0

  uint BdmType: Use PXE boot when set to 0, BDM (Bios) boot when set to
                  1 and BDM (Uefi) boot when set to 2. Default=0

bdmFormat   1 use VHD for BDMboot, 2 use ISO, 3 use USB.
                         Default=0

  uint BdmFormat: 1 use VHD for BDMboot, 2 use ISO, 3 use USB.
                  Default=0

bdmUpdated   Timestamp of the last BDM boot disk update.
                         Default=null

  DateTime BdmUpdated: Timestamp of the last BDM boot disk update.
                  Default=Empty

bdmCreated   Timstamp when BDM device was
                         created Default=null

  DateTime BdmCreated: Timstamp when BDM device was
                  created Default=Empty

```
xsPvsProxyUuid              UUID of XenServer PVS_proxy Default=null
                            Length=36
```

Guid XsPvsProxyUuid: UUID of XenServer PVS_proxy Default=00000000-
            0000-0000-0000-000000000000

```
enableXsProxy               Enable XenServerProxy when set to 1
                            Default=null
```

string EnableXsProxy: Enable XenServerProxy when set to 1 Default=""

Mcli-Set Disk

```
Set one or more field values for a Disk.
```

Set-PvsDisk

Set Disk(s) changed values from PvsDisk object(s), or set one or more field
   values for a PvsDisk.

  -p   Parameters needed to identify the record to
                            be changed.

When Disk is not passed the parameters below are used:

```
This required
```

```
diskLocatorId               GUID of the Disk Locator.
```

Guid Guid or DiskLocatorId: GUID of the Disk Locator.

```
or this required & resolution
```

```
diskLocatorName             Name of the Disk Locator.
```

string Name or DiskLocatorName: Name of the Disk Locator.

```
One of these resolutions when needed
```

```
siteId                      GUID of the Site.
```

Guid SiteId: GUID of the Site.

```
siteName                    Name of the Site.
```

string SiteName: Name of the Site.

```
One of these resolutions when needed
```

```
storeId                     GUID of the Store that is needed when a
                            diskLocatorName is used.
```

Guid StoreId: GUID of the Store that is needed when a DiskLocatorName
            is used.

```
storeName                   Name of the Store that is needed when a
                            diskLocatorName is used.
```

string StoreName: Name of the Store that is needed when a
            DiskLocatorName is used.

Required

PvsDisk[] Disk: Array of PvsDisk objects with changed property
            value(s) to be set. The object(s) can come from a
            pileline.

Optional

SwitchParameter PassThru: If -PassThru is specified, the resulting
PvsDisk object(s) are returned.

SwitchParameter Confirm: The impact of this operation is "low". If -
Confirm is specified, the operation will be
confirmed. $ConfirmPreference can be set to "low" to
have confirmation without the Confirm parameter.

-r   Record containing the new values for fields
to be changed.

Fields that can have value set

class   Class of the Disk. Max Length=40

string Class: Class of the Disk. Max Length=40

imageType   Type of this image (software type). Max
Length=40

string ImageType: Type of this image (software type). Max Length=40

writeCacheSize   RAM cache size (MB). Not 0 when used with
Cache in Device RAM, and Cache in Device RAM
with Overflow on Hard Disk. A value of 0
will disable the RAM use for Cache in Device
RAM with Overflow on Hard Disk. Min=0,
Max=131072, Default=0

UInt64 WriteCacheSize: RAM cache size (MB). Not 0 when used with
Cache in Device RAM, and Cache in Device RAM with
Overflow on Hard Disk. A value of 0 will disable the
RAM use for Cache in Device RAM with Overflow on
Hard Disk. Min=0, Max=131072, Default=0

autoUpdateEnabled   Automatically update this image for matching
Devices when set to 1. Default 0

bool AutoUpdateEnabled: Automatically update this image for matching
Devices when set to true. Default false

activationDateEnabled   Use activation date to activate image when
set to 1. Default 0

bool ActivationDateEnabled: Use activation date to activate image
when set to true. Default false

adPasswordEnabled   Enable AD password management when set to 1.

bool AdPasswordEnabled: Enable AD password management when set to
true.

haEnabled   Enable HA when set to 1.

bool HaEnabled: Enable HA when set to true.

printerManagementEnabled Invalid printers will be deleted from the
Device when set to 1.

bool PrinterManagementEnabled: Invalid printers will be deleted from
the Device when set to true.

writeCacheType   0 (Private), (other values are standard
image) 1 (Cache on Server), 3 (Cache in
Device RAM), 4 (Cache on Device Hard Disk),
7 (Cache on Server, Persistent), or 9 (Cache

in Device RAM with Overflow on Hard Disk).
Min=0, Max=9, Default=0

uint WriteCacheType: 0 (Private), (other values are standard image) 1
(Cache on Server), 3 (Cache in Device RAM), 4 (Cache
on Device Hard Disk), 7 (Cache on Server,
Persistent), or 9 (Cache in Device RAM with Overflow
on Hard Disk). Min=0, Max=9, Default=0

licenseMode   0 (None), 1 (Multiple Activation Key), or 2
                           (Key Management Service). Min=0, Max=2,
                           Default=0

uint LicenseMode: 0 (None), 1 (Multiple Activation Key), or 2 (Key
Management Service). Min=0, Max=2, Default=0

activeDate   Date to activate the disk if
                           autoUpdateEnabled and activationDateEnabled
                           are 1. Format is yyyy/mm/dd. Empty when the
                           autoUpdateEnabled or activationDateEnabled
                           are 0.

DateTime ActiveDate: Date to activate the disk if AutoUpdateEnabled
and activationDateEnabled are true. Has the date.
Empty when the AutoUpdateEnabled or
activationDateEnabled are false.

longDescription   Description of the Disk. Max Length=399

string LongDescription: Description of the Disk. Max Length=399

operatingSystem   Operating System of Disk. Max Length=250

string OperatingSystem: Operating System of Disk. Max Length=250

osType   Operating System Type of Disk. Max Length=40

string OsType: Operating System Type of Disk. Max Length=40

serialNumber   User defined serial number. Max Length=36

string SerialNumber: User defined serial number. Max Length=36

date   User defined date. Max Length=40

string Date: User defined date. Max Length=40

author   User defined author. Max Length=40

string Author: User defined author. Max Length=40

title   User defined title. Max Length=40

string Title: User defined title. Max Length=40

company   User defined company. Max Length=40

string Company: User defined company. Max Length=40

internalName   User defined name. Max Length=63

string InternalName: User defined name. Max Length=63

originalFile   User defined original file. Max Length=127

string OriginalFile: User defined original file. Max Length=127

hardwareTarget   User defined hardware target. Max Length=127

string HardwareTarget: User defined hardware target. Max Length=127

    majorRelease   User defined major release number. Min=0,
                                    Max=4294967295, Default=0

     UInt64 MajorRelease: User defined major release number. Min=0,
                 Max=4294967295, Default=0

    minorRelease   User defined minor release number. Min=0,
                                    Max=4294967295, Default=0

     UInt64 MinorRelease: User defined minor release number. Min=0,
                 Max=4294967295, Default=0

    build   User defined build number. Min=0,
                                    Max=4294967295, Default=0

     UInt64 Build: User defined build number. Min=0, Max=4294967295,
                 Default=0

    clearCacheDisabled   Clear cached secrets disabled.

     string ClearCacheDisabled: Clear cached secrets disabled.

    format   Format of the image. Values are: 0 (VHD), 1
                                    (VHDX). Default=0

     bool VHDX: If VHDX is true, the format of the image is VHDX.
                 Otherwise it is VHD. Default=false

Mcli-Set DiskLocator

    Set one or more field values for a Disk Locator.

Set-PvsDiskLocator

    Set Disk Locator(s) changed values from PvsDiskLocator object(s), or set
     one or more field values for a PvsDiskLocator.

    -p   Parameters needed to identify the record to
                                    be changed.

     When DiskLocator is not passed the parameters below are used:
    This required
     diskLocatorId   GUID of the Disk Locator to Set.

       Guid Guid or DiskLocatorId: GUID of the Disk Locator to Set.

    or this required & resolution
     diskLocatorName   Name of the Disk Locator File to Set.

       string Name or DiskLocatorName: Name of the Disk Locator File to Set.

    One of these resolutions when needed
     siteId   GUID of the Site.

       Guid SiteId: GUID of the Site.

     siteName   Name of the Site.

       string SiteName: Name of the Site.

    One of these resolutions when needed

storeId   GUID of the Store that is needed when a
                           diskLocatorName is used.

  Guid StoreId: GUID of the Store that is needed when a DiskLocatorName
                is used.

storeName   Name of the Store that is needed when a
                           diskLocatorName is used.

  string StoreName: Name of the Store that is needed when a
                DiskLocatorName is used.

Required

  PvsDiskLocator[] DiskLocator: Array of PvsDiskLocator objects with
                changed property value(s) to be set. The object(s)
                can come from a pileline.

Optional

  SwitchParameter PassThru: If -PassThru is specified, the resulting
                PvsDiskLocator object(s) are returned.

  SwitchParameter Confirm: The impact of this operation is "low". If -
                Confirm is specified, the operation will be
                confirmed. $ConfirmPreference can be set to "low" to
                have confirmation without the Confirm parameter.

-r   Record containing the new values for fields
                           to be changed.

 Fields that can have value set

  description   User description. Default="" Max Length=250

   string Description: User description. Default="" Max Length=250

  menuText   Text that is displayed in the Boot Menu. If
                           this field has no value, the name value is
                           used. Default="" ASCII Max Length=64

   string MenuText: Text that is displayed in the Boot Menu. If this
                field has no value, the name value is used.
                Default="" ASCII Max Length=64

  serverId   GUID of the single Server that this Disk
                           Locator is assigned to. It is not used with
                           serverName. Default=null Length=36

   Guid ServerId: GUID of the single Server that this Disk Locator is
                assigned to. It is not used with ServerName.
                Default=00000000-0000-0000-0000-000000000000

  serverName   Name of the single Server that this Disk
                           Locator is assigned to. It is not used with
                           serverId. Default=null

   string ServerName: Name of the single Server that this Disk Locator
                is assigned to. It is not used with ServerId.
                Default=""

  enabled   1 when this disk can be booted, 0 otherwise.
                           Default=1

   bool Enabled: True when this disk can be booted, false otherwise.
                Default=true

```
rebalanceEnabled           1 when this Server can automatically
                           rebalance Devices, 0 otherwise. Default=0
```

bool RebalanceEnabled: True when this Server can automatically
                       rebalance Devices, false otherwise. Default=false

```
rebalanceTriggerPercent   Percent over fair load that triggers a
                          dynamic Device rebalance. Min=5, Max=5000,
                          Default=25
```

uint RebalanceTriggerPercent: Percent over fair load that triggers a
                       dynamic Device rebalance. Min=5, Max=5000,
                       Default=25

```
subnetAffinity            Qualifier for subnet affinity when assigning
                          a Server. 0=None, 1=Best Effort, 2=Fixed.
                          Min=0, Max=2, Default=0
```

uint SubnetAffinity: Qualifier for subnet affinity when assigning a
                     Server. 0=None, 1=Best Effort, 2=Fixed. Min=0,
                     Max=2, Default=0

Mcli-Set DiskUpdateDevice

```
Set one or more field values for one or more Disk Update Devices.
```

Set-PvsDiskUpdateDevice

Set Disk Update Device(s) changed values from PvsDiskUpdateDevice
   object(s), or set one or more field values for one or more
   PvsDiskUpdateDevices.

```
  -p                        Parameters needed to identify the record to
                            be changed.
```

When DiskUpdateDevice is not passed the parameters below are used:
```
  One of these required
```
```
  deviceId                GUID of the Disk Update Device to Set.
```
   Guid Guid or DeviceId: GUID of the Disk Update Device to Set.
```
  deviceName              Name of Disk Update Device to Set.
```
   string Name or DeviceName: Name of Disk Update Device to Set.
```
  deviceMac               MAC of Disk Update Device to Set.
```
   PvsPhysicalAddress DeviceMac: MAC of Disk Update Device to Set.
```
  siteId                  GUID of the Site. Can be used alone to Set
                          all Disk Update Devices in the Site.
```
   Guid SiteId: GUID of the Site. Can be used alone to Set all Disk
                Update Devices in the Site.
```
  siteName                Name of the Site. Can be used alone to Set
                          all Disk Update Devices in the Site.
```
   string SiteName: Name of the Site. Can be used alone to Set all Disk
                    Update Devices in the Site.
```
  diskLocatorId           GUID of the DiskLocator to Set the Disk
                          Update Device for.
```

Guid DiskLocatorId: GUID of the DiskLocator to Set the Disk Update
                  Device for.
  or this required & resolution

  diskLocatorName   Name of the DiskLocator to Set the Disk
                            Update Device for.

    string DiskLocatorName: Name of the DiskLocator to Set the Disk
                  Update Device for.

One of these resolutions when needed

  siteId   GUID of the Site.

    Guid SiteId: GUID of the Site. Can be used alone to Set all Disk
                  Update Devices in the Site.

  siteName   Name of the Site.

    string SiteName: Name of the Site. Can be used alone to Set all Disk
                  Update Devices in the Site.

One of these resolutions when needed

  storeId   GUID of the Store that is needed when a
                            diskLocatorName is used.

    Guid StoreId: GUID of the Store that is needed when a DiskLocatorName
                  is used.

  storeName   Name of the Store that is needed when a
                            diskLocatorName is used.

    string StoreName: Name of the Store that is needed when a
                  DiskLocatorName is used.

  Required

    PvsDiskUpdateDevice[] DiskUpdateDevice: Array of PvsDiskUpdateDevice
                  objects with changed property value(s) to be set.
                  The object(s) can come from a pileline.

  Optional

    SwitchParameter PassThru: If -PassThru is specified, the resulting
                  PvsDiskUpdateDevice object(s) are returned.

    SwitchParameter Confirm: The impact of this operation is "low". If -
                  Confirm is specified, the operation will be
                  confirmed. $ConfirmPreference can be set to "low" to
                  have confirmation without the Confirm parameter.

  -r   Record containing the new values for fields
                            to be changed.

  Fields that can have value set

  description   User description. Default="" Max Length=250

    string Description: User description. Default="" Max Length=250

  port   UDP port to use with Stream Service.
                            Min=1025, Max=65534, Default=6901

    uint Port: UDP port to use with Stream Service. Min=1025, Max=65534,
                  Default=6901

adTimestamp   The time the Active Directory machine
                               account password was generated. Do not set
                               this field, it is only set internally by
                               PVS. Default=0

 uint AdTimestamp: The time the Active Directory machine account
                   password was generated. Do not set this field, it is
                   only set internally by PVS. Default=0

adSignature   The signature of the Active Directory
                               machine account password. Do not set this
                               field, it is only set internally by PVS.
                               Default=0

 uint AdSignature: The signature of the Active Directory machine
                   account password. Do not set this field, it is only
                   set internally by PVS. Default=0

logLevel   Level to perform logging at. Values are: 0
                               (None), 1 (Fatal), 2 (Error), 3 (Warning), 4
                               (Info), 5 (Debug), and 6 (Trace). Min=0,
                               Max=6, Default=0

 uint LogLevel: Level to perform logging at. Values are: 0 (None), 1
                (Fatal), 2 (Error), 3 (Warning), 4 (Info), 5
                (Debug), and 6 (Trace). Min=0, Max=6, Default=0

domainName   Fully qualified name of the domain that the
                               Device belongs to. Do not set this field, it
                               is only set internally by PVS. Default=null
                               Max Length=255

 string DomainName: Fully qualified name of the domain that the Device
                    belongs to. Do not set this field, it is only set
                    internally by PVS. Default="" Max Length=255

domainObjectSID   The value of the objectSID AD attribute of
                               the same name for the Device's computer
                               account. Do not set this field, it is only
                               set internally by PVS. Default=null Max
                               Length=186

 string DomainObjectSID: The value of the objectSID AD attribute of
                    the same name for the Device's computer account. Do
                    not set this field, it is only set internally by
                    PVS. Default="" Max Length=186

domainControllerName   The name of the DC used to create the host's
                               computer account. Do not set this field, it
                               is only set internally by PVS. Default=null
                               Max Length=4000

 string DomainControllerName: The name of the DC used to create the
                    host's computer account. Do not set this field, it
                    is only set internally by PVS. Default="" Max
                    Length=4000

domainTimeCreated   The time that the computer account was
                               created. Do not set this field, it is only
                               set internally by PVS. Format is: YYYY-MM-DD
                               HH:MM:SS.SSS Default=null

adPassword   The Active Directory machine account password. Do not set this field, it is only set internally by PVS. Default="" Max Length=256

Mcli-Set DiskVersion

Set one or more field values for one Disk Version.

Set-PvsDiskVersion

Set Disk Version(s) changed values from PvsDiskVersion object(s), or set one or more field values for a PvsDiskVersion.

-p   Parameters needed to identify the record to be changed.

When DiskVersion is not passed the parameters below are used:

This required

diskLocatorId   GUID of the Disk Locator Version to Set.

Guid Guid or DiskLocatorId: GUID of the Disk Locator Version to Set.

or this required & resolution

diskLocatorName   Name of the Disk Locator Version to Set.

string Name or DiskLocatorName: Name of the Disk Locator Version to Set.

This required

version   Version to Set.

uint Version: Version to Set.

One of these resolutions when needed

siteId   GUID of the Site.

Guid SiteId: GUID of the Site.

siteName   Name of the Site.

string SiteName: Name of the Site.

One of these resolutions when needed

storeId   GUID of the Store that is needed when a diskLocatorName is used.

Guid StoreId: GUID of the Store that is needed when a DiskLocatorName is used.

storeName   Name of the Store that is needed when a diskLocatorName is used.

```
            string StoreName: Name of the Store that is needed when a
                        DiskLocatorName is used.

        Required

          PvsDiskVersion[] DiskVersion: Array of PvsDiskVersion objects with
                        changed property value(s) to be set. The object(s)
                        can come from a pileline.

        Optional

          SwitchParameter PassThru: If -PassThru is specified, the resulting
                        PvsDiskVersion object(s) are returned.

          SwitchParameter Confirm: The impact of this operation is "low". If -
                        Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "low" to
                        have confirmation without the Confirm parameter.

    -r                          Record containing the new values for fields
                            to be changed.

      Fields that can have value set

      description                   User description. Default="" Max Length=250

        string Description: User description. Default="" Max Length=250

      scheduledDate                 Date/Time that the Disk Version is scheduled
                                to become available. Format is: YYYY-MM-DD
                                HH:MM. When empty, the disk version is made
                                available immediately. Default=null

        DateTime ScheduledDate: Date/Time that the Disk Version is scheduled
                        to become available. Has the date, hour and minute.
                        Empty when the disk version is made available
                        immediately. Default=Empty
```

Mcli-Set Farm

```
    Set one or more field values for a Farm.
```

Set-PvsFarm

```
    Set Farm changed values from a PvsFarm object, or set one or more field
      values for a PvsFarm.

      -p                            Parameters needed to identify the record to
                                be changed.

        When Farm is not passed the parameters below are used:

        Optional

         farmId                     GUID of the Farm to Set. This is optional
                                    since there is only one Farm.

           Guid Guid or FarmId: GUID of the Farm to Set. This is optional since
                        there is only one Farm.

          Required

           PvsFarm[] Farm: Array of PvsFarm objects with changed property
                        value(s) to be set. The object(s) can come from a
                        pileline.
```

SwitchParameter PassThru: If -PassThru is specified, the resulting
                    PvsFarm object is returned.

SwitchParameter Confirm: The impact of this operation is "low". If -
                    Confirm is specified, the operation will be
                    confirmed. $ConfirmPreference can be set to "low" to
                    have confirmation without the Confirm parameter.

-r   Record containing the new values for fields
                          to be changed.

 Fields that can have value set

farmName   Name of the Farm. Default="" Max Length=50

  string NewName: Name of the Farm. Default="" Max Length=50

description   User description. Default="" Max Length=250

  string Description: User description. Default="" Max Length=250

autoAddEnabled   1 when Auto Add is enabled, 0 otherwise.
                          Default=0

 bool AutoAddEnabled: True when Auto Add is enabled, false otherwise.
                    Default=false

auditingEnabled   1 when Auditing is enabled, 0 otherwise.
                          Default=0

 bool AuditingEnabled: True when Auditing is enabled, false otherwise.
                    Default=false

lastAuditArchiveDate   Last date of Audit Trail data that was
                          Archived. Format is yyyy/mm/dd Default=null

  DateTime LastAuditArchiveDate: Last date of Audit Trail data that was
                    Archived. Has the date. Default=Empty

defaultSiteId   GUID of the Site to place new Devices into
                          automatically. Not used with
                          defaultSiteName. Default=null Length=36

  Guid DefaultSiteId: GUID of the Site to place new Devices into
                    automatically. Not used with defaultSiteName.
                    Default=00000000-0000-0000-0000-000000000000

defaultSiteName   Name of the Site to place new Devices into
                          automatically. Not used with defaultSiteId.
                          Default=null

  string DefaultSiteName: Name of the Site to place new Devices into
                    automatically. Not used with DefaultSiteId.
                    Default=""

offlineDatabaseSupportEnabled 1 when Offline Database Support is
                          enabled, 0 otherwise. Default=0

 bool OfflineDatabaseSupportEnabled: True when Offline Database
                    Support is enabled, false otherwise. Default=false

licenseServer   License server name. Default="" Max
                          Length=255

  string LicenseServer: License server name. Default="" Max Length=255

```
licenseServerPort          License server port. Min=1025, Max=65534,
                           Default=27000

     uint LicenseServerPort: License server port. Min=1025, Max=65534,
                  Default=27000

licenseTradeUp             License server trade up, when set to 1.
                           Default=0

     bool LicenseTradeUp: License server trade up, when set to true.
                  Default=false

automaticMergeEnabled      1 when Automatic Merge is enabled, 0
                           otherwise. If the number of versions becomes
                           more than the maxVersions value, a merge
                           will occur at the end of PromoteDiskVersion.
                           Default=1

     bool AutomaticMergeEnabled: True when Automatic Merge is enabled,
                  false otherwise. If the number of versions becomes
                  more than the MaxVersions value, a merge will occur
                  at the end of PromoteDiskVersion. Default=true

maxVersions                Maximum number a versions of a Disk that can
                           exist before a merge will automatically
                           occur. Min=3, Max=50, Default=5

     uint MaxVersions: Maximum number a versions of a Disk that can exist
                  before a merge will automatically occur. Min=3,
                  Max=50, Default=5

mergeMode                  Mode to place the version in after a merge
                           has occurred. Values are: 0 (Production), 1
                           (Test) and 2 (Maintenance). Min=0, Max=2,
                           Default=2

     uint MergeMode: Mode to place the version in after a merge has
                  occurred. Values are: 0 (Production), 1 (Test) and 2
                  (Maintenance). Min=0, Max=2, Default=2
```

Mcli-Set FarmView

```
Set one or more field values for a Farm View.
```

Set-PvsFarmView

```
Set Farm View(s) changed values from PvsFarmView object(s), or set one or
   more field values for a PvsFarmView.

  -p                       Parameters needed to identify the record to
                           be changed.

   When FarmView is not passed the parameters below are used:

  One of these required

  farmViewId               GUID of the Farm View to Set.

     Guid Guid or FarmViewId: GUID of the Farm View to Set.

  farmViewName             Name of the Farm View to Set.

     string Name or FarmViewName: Name of the Farm View to Set.

   Required
```

```
            PvsFarmView[] FarmView: Array of PvsFarmView objects with changed
                          property value(s) to be set. The object(s) can come
                          from a pileline.

        Optional

          SwitchParameter PassThru: If -PassThru is specified, the resulting
                          PvsFarmView object(s) are returned.

          SwitchParameter Confirm: The impact of this operation is "low". If -
                          Confirm is specified, the operation will be
                          confirmed. $ConfirmPreference can be set to "low" to
                          have confirmation without the Confirm parameter.

   -r                              Record containing the new values for fields
                                   to be changed.

    Fields that can have value set

     farmViewName                 name of the Farm View. Max Length=50

       string NewName: name of the Farm View. Max Length=50

     description                  User description. Default="" Max Length=250

       string Description: User description. Default="" Max Length=250
```

Mcli-Set Server

```
   Set one or more field values for a Server. Restart the Stream Service after
      setting the fields.
```

Set-PvsServer

```
   Set Server(s) changed values from PvsServer object(s), or set one or more
      field values for a PvsServer. Restart the Server(s) after setting.

   -p                              Parameters needed to identify the record to
                                   be changed.

     When Server is not passed the parameters below are used:

    One of these required

     serverId                     GUID of the Server to Set.

       Guid Guid or ServerId: GUID of the Server to Set.

     serverName                   Name of the Server to Set.

       string Name or ServerName: Name of the Server to Set.

      Required

        PvsServer[] Server: Array of PvsServer objects with changed property
                        value(s) to be set. The object(s) can come from a
                        pileline.

        Optional

          SwitchParameter PassThru: If -PassThru is specified, the resulting
                          PvsServer object(s) are returned.

          SwitchParameter Confirm: The impact of this operation is "low". If -
                          Confirm is specified, the operation will be
                          confirmed. $ConfirmPreference can be set to "low" to
                          have confirmation without the Confirm parameter.
```

```
-r                              Record containing the new values for fields
                                to be changed.
```

Fields that can have value set

```
serverName                      Computer name with no spaces. ASCII computer
                                name characters Max Length=21
```

string NewName: Computer name with no spaces. ASCII computer name
characters Max Length=21

```
description                     User description. Default="" Max Length=250
```

string Description: User description. Default="" Max Length=250

```
adMaxPasswordAge                Number of days before a password expires.
                                Min=1, Max=30, Default=7
```

uint AdMaxPasswordAge: Number of days before a password expires.
Min=1, Max=30, Default=7

```
licenseTimeout                  Amount of seconds before a license times
                                out. Min=15, Max=300, Default=30
```

uint LicenseTimeout: Amount of seconds before a license times out.
Min=15, Max=300, Default=30

```
vDiskCreatePacing               VDisk create time pacing in miliseconds.
                                Min=0, Max=5, Default=0
```

uint VDiskCreatePacing: VDisk create time pacing in miliseconds.
Min=0, Max=5, Default=0

```
firstPort                       Number of the first UDP port for use by the
                                Stream Service, First and Last must allow at
                                least 5 ports. Min=1025, Max=65534,
                                Default=6910
```

uint FirstPort: Number of the first UDP port for use by the Stream
Service, First and Last must allow at least 5 ports.
Min=1025, Max=65534, Default=6910

```
lastPort                        Number of the last UDP port for use by the
                                Stream Service, First and Last must allow at
                                least 5 ports. Min=1025, Max=65534,
                                Default=6930
```

uint LastPort: Number of the last UDP port for use by the Stream
Service, First and Last must allow at least 5 ports.
Min=1025, Max=65534, Default=6930

```
threadsPerPort                  Number of worker threads per IO port.
                                Required that (threadPerPort * numberPorts *
                                numberIPs) <= 1000. Min=1, Max=60, Default=8
```

uint ThreadsPerPort: Number of worker threads per IO port. Required
that (threadPerPort * numberPorts * numberIPs) <=
1000. Min=1, Max=60, Default=8

```
buffersPerThread                Number of buffers per worker thread. Min=1,
                                Max=128, Default=24
```

uint BuffersPerThread: Number of buffers per worker thread. Min=1,
Max=128, Default=24

serverCacheTimeout   Number of seconds to wait before considering
                            another Server is down. Min=5, Max=60,
                            Default=8

  uint ServerCacheTimeout: Number of seconds to wait before considering
                     another Server is down. Min=5, Max=60, Default=8

ioBurstSize   Number of bytes read/writes can send in a
                            burst of packets. Required that
                            ioBurstSize/(maxTransmissionUnits-76)<=32.
                            Min=4096, Max=61440, Default=32768

  uint IoBurstSize: Number of bytes read/writes can send in a burst of
                     packets. Required that
                     IoBurstSize/(MaxTransmissionUnits-76)<=32. Min=4096,
                     Max=61440, Default=32768

maxTransmissionUnits   Ethernet maximum transmission unit size for
                            the protocol for use for Server and Device.
                            Required that
                            ioBurstSize/(maxTransmissionUnits-76)<=32.
                            Min=502, Max=16426, Default=1506

  uint MaxTransmissionUnits: Ethernet maximum transmission unit size
                     for the protocol for use for Server and Device.
                     Required that IoBurstSize/(MaxTransmissionUnits-
                     76)<=32. Min=502, Max=16426, Default=1506

maxBootDevicesAllowed   Maximum number of Devices allowed to boot
                            simultaneously. Min=1, Max=1000, Default=500

  uint MaxBootDevicesAllowed: Maximum number of Devices allowed to boot
                     simultaneously. Min=1, Max=1000, Default=500

maxBootSeconds   Maximum number of seconds for a Device to
                            boot. Min=10, Max=900, Default=60

  uint MaxBootSeconds: Maximum number of seconds for a Device to boot.
                     Min=10, Max=900, Default=60

bootPauseSeconds   Number of seconds that a Device will pause
                            during login if its server busy. Min=1,
                            Max=60, Default=10

  uint BootPauseSeconds: Number of seconds that a Device will pause
                     during login if its server busy. Min=1, Max=60,
                     Default=10

adMaxPasswordAgeEnabled   Age the password, when set to 1. Default=0

  bool AdMaxPasswordAgeEnabled: Age the password, when set to true.
                     Default=false

eventLoggingEnabled   Enable event logging, when set to 1.
                            Default=0

  bool EventLoggingEnabled: Enable event logging, when set to true.
                     Default=false

nonBlockingIoEnabled   Use non-Blocking IO, when set to 1.
                            Default=1

  bool NonBlockingIoEnabled: Use non-Blocking IO, when set to true.
                     Default=true

ip   One or more streaming ip addresses. If more
                             than one ip is included, the ip addresses
                             are comma delimited.

  string[] Ip: One or more streaming ip addresses.

initialQueryConnectionPoolSize Initial size of database connection
                             pool for non-transactional queries. Min=1,
                             Max=1000, Default=50

  uint InitialQueryConnectionPoolSize: Initial size of database
                   connection pool for non-transactional queries.
                   Min=1, Max=1000, Default=50

initialTransactionConnectionPoolSize Initial size of database
                             connection pool for transactional queries.
                             Min=1, Max=1000, Default=50

  uint InitialTransactionConnectionPoolSize: Initial size of database
                   connection pool for transactional queries. Min=1,
                   Max=1000, Default=50

maxQueryConnectionPoolSize Maximum size of database connection pool
                             for non-transactional queries. Min=1,
                             Max=32767, Default=1000

  uint MaxQueryConnectionPoolSize: Maximum size of database connection
                   pool for non-transactional queries. Min=1,
                   Max=32767, Default=1000

maxTransactionConnectionPoolSize Maximum size of database connection
                             pool for transactional queries. Min=1,
                             Max=32767, Default=1000

  uint MaxTransactionConnectionPoolSize: Maximum size of database
                   connection pool for transactional queries. Min=1,
                   Max=32767, Default=1000

refreshInterval   Interval, in number of seconds, the server
                             should wait before refreshing settings. If
                             set to 0, unused database connections are
                             never released. Min=0, Max=32767,
                             Default=300

  uint RefreshInterval: Interval, in number of seconds, the server
                   should wait before refreshing settings. If set to 0,
                   unused database connections are never released.
                   Min=0, Max=32767, Default=300

unusedDbConnectionTimeout   Interval, in number of seconds, a
                             connection should go unused before it is to
                             be released. Min=0, Max=32767, Default=300

  uint UnusedDbConnectionTimeout: Interval, in number of seconds, a
                   connection should go unused before it is to be
                   released. Min=0, Max=32767, Default=300

busyDbConnectionRetryCount   Number of times a failed database
                             connection will be retried. Min=0,
                             Max=32767, Default=2

  uint BusyDbConnectionRetryCount: Number of times a failed database
                   connection will be retried. Min=0, Max=32767,
                   Default=2

```
busyDbConnectionRetryInterval  Interval, in number of milliseconds,
                               the server should wait before retrying to
                               connect to a database. Min=0, Max=10000,
                               Default=25
```

 uint BusyDbConnectionRetryInterval: Interval, in number of
                 milliseconds, the server should wait before retrying
                 to connect to a database. Min=0, Max=10000,
                 Default=25

```
localConcurrentIoLimit     Maximum concurrent IO transactions it
                           performs for vDisks that are local. A value
                           of 0 disables the feature. Min=0, Max=128,
                           Default=4
```

 uint LocalConcurrentIoLimit: Maximum concurrent IO transactions it
                 performs for vDisks that are local. A value of 0
                 disables the feature. Min=0, Max=128, Default=4

```
remoteConcurrentIoLimit    Maximum concurrent IO transactions it
                           performs for vDisks that are remote. A value
                           of 0 disables the feature. Min=0, Max=128,
                           Default=4
```

 uint RemoteConcurrentIoLimit: Maximum concurrent IO transactions it
                 performs for vDisks that are remote. A value of 0
                 disables the feature. Min=0, Max=128, Default=4

```
logLevel                   Level to perform logging at. Values are: 0
                           (None), 1 (Fatal), 2 (Error), 3 (Warning), 4
                           (Info), 5 (Debug), and 6 (Trace). Min=0,
                           Max=6, Default=4
```

 uint LogLevel: Level to perform logging at. Values are: 0 (None), 1
                 (Fatal), 2 (Error), 3 (Warning), 4 (Info), 5
                 (Debug), and 6 (Trace). Min=0, Max=6, Default=4

```
logFileSizeMax             Maximum size log files can reach in
                           Megabytes. Min=1, Max=50, Default=5
```

 uint LogFileSizeMax: Maximum size log files can reach in Megabytes.
                 Min=1, Max=50, Default=5

```
logFileBackupCopiesMax     Maximum number of log file backups. Min=1,
                           Max=50, Default=4
```

 uint LogFileBackupCopiesMax: Maximum number of log file backups.
                 Min=1, Max=50, Default=4

```
powerRating                A strictly relative rating of this Server's
                           capabilities when compared to other Servers
                           in the Store(s) it belongs too; can be used
                           to help tune load balancing. Min=0.1,
                           Max=1000, Default=1
```

 float PowerRating: A strictly relative rating of this Server's
                 capabilities when compared to other Servers in the
                 Store(s) it belongs too; can be used to help tune
                 load balancing. Min=0.1, Max=1000, Default=1

```
lastCeipUploadAttempt      Time that this server last attempted a CEIP
                           upload. Default=null
```

lastBugReportAttempt   Time that this server last attempted to upload or generate a bug report bundle. Default=null

lastBugReportStatus   Status of the last bug report on this server. Default=null Max Length=250

lastBugReportResult   Status of the last bug report on this server. Default=null Max Length=4000

lastBugReportSummary   Summary of the last bug report on this server. Default=null Max Length=250

string LastBugReportSummary: Summary of the last bug report on this server. Default="" Max Length=250

Mcli-Set ServerBiosBootstrap

Oem Only: Set the bootstrap fields for the Server dell_bios.bin BIOS bootstrap file.

Set-PvsServerBiosBootstrap

Oem Only: Set Server Bios Bootstrap(s) changed values from PvsServerBiosBootstrap object(s), or set one or more field values for a PvsServerBiosBootstrap.

-p   Parameters needed to identify the record to be changed.

When ServerBiosBootstrap is not passed the parameters below are used:

One of these required

serverId   GUID of the Server to Set the dell_bios.bin BIOS bootstrap file.

Guid Guid or ServerId: GUID of the Server to Set the dell_bios.bin BIOS bootstrap file.

serverName   Name of the Server to Set the dell_bios.bin BIOS bootstrap file.

string Name or ServerName: Name of the Server to Set the dell_bios.bin BIOS bootstrap file.

Required

PvsServerBiosBootstrap[] ServerBiosBootstrap: Array of PvsServerBiosBootstrap objects with changed property value(s) to be set. The object(s) can come from a pileline.

SwitchParameter PassThru: If -PassThru is specified, the resulting
PvsServerBiosBootstrap object(s) are returned.

SwitchParameter Confirm: The impact of this operation is "low". If -
Confirm is specified, the operation will be
confirmed. $ConfirmPreference can be set to "low" to
have confirmation without the Confirm parameter.

-r   Record containing the new values for fields
to be changed.

Fields that can have value set

enabled   Automatically update the BIOS on the target
device with these setting when set to 1,
otherwise do not use these settings.
Default=0

bool Enabled: Automatically update the BIOS on the target device with
these setting when set to true, otherwise do not use
these settings. Default=false

dhcpEnabled   Use DHCP to retrieve target device IP when
set to 1, otherwise use the static domain,
dnsIpAddress1 and dnsIpAddress2 settings.
Default=1

bool DhcpEnabled: Use DHCP to retrieve target device IP when set to
true, otherwise use the static domain,
dnsIpAddresstrue and dnsIpAddress2 settings.
Default=true

lookup   Use DNS to find the Server when set to 1
with the serverName host value, otherwise
use the bootserver1_Ip, bootserver1_Port,
bootserver2_Ip, bootserver2_Port,
bootserver3_Ip, bootserver3_Port,
bootserver4_Ip, and bootserver4_Port
settings. Default=1

bool Lookup: Use DNS to find the Server when set to true with the
ServerName host value, otherwise use the
bootservertrue_Ip, bootservertrue_Port,
bootserver2_Ip, bootserver2_Port, bootserver3_Ip,
bootserver3_Port, bootserver4_Ip, and
bootserver4_Port settings. Default=true

verboseMode   Display verbose diagnostic information when
set to 1. Default=0

bool VerboseMode: Display verbose diagnostic information when set to
true. Default=false

interruptSafeMode   Interrupt safe mode (use if target device
hangs during boot) when set to 1. Default=0

bool InterruptSafeMode: Interrupt safe mode (use if target device
hangs during boot) when set to true. Default=false

paeMode   PAE mode (use if PAE enabled in boot.ini of
target device) when set to 1. Default=0

bool PaeMode: PAE mode (use if PAE enabled in boot.ini of target device) when set to true. Default=false

bootFromHdOnFail   For network recovery reboot to hard drive when set to 1, restore network connection when set to 0. Default=0

bool BootFromHdOnFail: For network recovery reboot to hard drive when set to true, restore network connection when set to false. Default=false

recoveryTime   When bootFromHdOnFail is 1, this is the number of seconds to wait before reboot to hard drive. Min=10, Max=60000, Default=50

uint RecoveryTime: When bootFromHdOnFail is 1, this is the number of seconds to wait before reboot to hard drive. Min=10, Max=60000, Default=50

pollingTimeout   Login polling timeout in milliseconds. Min=1000, Max=60000, Default=5000

uint PollingTimeout: Login polling timeout in milliseconds. Min=1000, Max=60000, Default=5000

generalTimeout   Login general timeout in milliseconds. Min=1000, Max=60000, Default=5000

uint GeneralTimeout: Login general timeout in milliseconds. Min=1000, Max=60000, Default=5000

serverName   Host to use for DNS lookup. Only used when lookup is 1. Default=IMAGESERVER1

string NewName: Host to use for DNS lookup. Only used when Lookup is true. Default=IMAGESERVER1

bootserver1_Ip   1st boot server IP. Only used when lookup is 0.

string Bootserver1_Ip: 1st boot server IP. Only used when Lookup is false.

bootserver1_Port   1st boot server port. Only used when lookup is 0. Min=1025, Max=65536, Default=6910

uint Bootserver1_Port: 1st boot server port. Only used when Lookup is false. Min=1025, Max=65536, Default=6910

bootserver2_Ip   2nd boot server IP. Only used when lookup is 0. Default=0.0.0.0

string Bootserver2_Ip: 2nd boot server IP. Only used when Lookup is false. Default=0.0.0.0

bootserver2_Port   2nd boot server port. Only used when lookup is 0. Min=1025, Max=65536, Default=6910

uint Bootserver2_Port: 2nd boot server port. Only used when Lookup is false. Min=1025, Max=65536, Default=6910

bootserver3_Ip   3rd boot server IP. Only used when lookup is 0. Default=0.0.0.0

string Bootserver3_Ip: 3rd boot server IP. Only used when Lookup is false. Default=0.0.0.0

```
        bootserver3_Port            3rd boot server port. Only used when lookup
                                    is 0. Min=1025, Max=65536, Default=6910

        uint Bootserver3_Port: 3rd boot server port. Only used when Lookup is
                        false. Min=1025, Max=65536, Default=6910

        bootserver4_Ip              4th boot server IP. Only used when lookup is
                                    0. Default=0.0.0.0

        string Bootserver4_Ip: 4th boot server IP. Only used when Lookup is
                        false. Default=0.0.0.0

        bootserver4_Port            4th boot server port. Only used when lookup
                                    is 0. Min=1025, Max=65536, Default=6910

        uint Bootserver4_Port: 4th boot server port. Only used when Lookup is
                        false. Min=1025, Max=65536, Default=6910

        domain                      Domain of the primary and secondary DNS
                                    servers. Only used when dhcpEnabled is 0.

        string Domain: Domain of the primary and secondary DNS servers. Only
                        used when DhcpEnabled is false.

        dnsIpAddress1               Primary DNS server IP. Only used when
                                    dhcpEnabled is 0.

        string DnsIpAddress1: Primary DNS server IP. Only used when
                        DhcpEnabled is false.

        dnsIpAddress2               Secondary DNS server IP. Only used when
                                    dhcpEnabled is 0.

        string DnsIpAddress2: Secondary DNS server IP. Only used when
                        DhcpEnabled is false.
```

Mcli-Set ServerBootstrap

```
    Set the bootstrap fields for the Server and named bootstrap file specified.
```

Set-PvsServerBootstrap

```
    Set Server Bootstrap(s) changed values from PvsServerBootstrap object(s),
       or set one or more field values for a PvsServerBootstarp.

      -p                            Parameters needed to identify the record to
                                    be changed.

        When ServerBootstrap is not passed the parameters below are used:

      One of these required

        serverId                    GUID of the Server to Set the named
                                    bootstrap file on.

        Guid Guid or ServerId: GUID of the Server to Set the named bootstrap
                        file on.

        serverName                  Name of the Server to Set the named
                                    bootstrap file on.

        string ServerName: Name of the Server to Set the named bootstrap file
                        on.

      This required

        name                        Name of the bootstrap file.
```

string Name: Name of the bootstrap file.

    Required

      PvsServerBootstrap[] ServerBootstrap: Array of PvsServerBootstrap
                    objects with changed property value(s) to be set.
                    The object(s) can come from a pileline.

    Optional

      SwitchParameter PassThru: If -PassThru is specified, the resulting
                    PvsServerBootstrap object(s) are returned.

      SwitchParameter Confirm: The impact of this operation is "low". If -
                    Confirm is specified, the operation will be
                    confirmed. $ConfirmPreference can be set to "low" to
                    have confirmation without the Confirm parameter.

  -r   Record containing the new values for fields
                                to be changed.

  Fields that can have value set

  bootserver1_Ip   1st boot server IP.

    string Bootserver1_Ip: 1st boot server IP.

  bootserver1_Netmask   1st boot server netmask. Default=0.0.0.0

    string Bootserver1_Netmask: 1st boot server netmask. Default=0.0.0.0

  bootserver1_Gateway   1st boot server gateway. Default=0.0.0.0

    string Bootserver1_Gateway: 1st boot server gateway. Default=0.0.0.0

  bootserver1_Port   1st boot server port. Min=1025, Max=65536,
                                Default=6910

    uint Bootserver1_Port: 1st boot server port. Min=1025, Max=65536,
                    Default=6910

  bootserver2_Ip   2nd boot server IP. Default=0.0.0.0

    string Bootserver2_Ip: 2nd boot server IP. Default=0.0.0.0

  bootserver2_Netmask   2nd boot server netmask. Default=0.0.0.0

    string Bootserver2_Netmask: 2nd boot server netmask. Default=0.0.0.0

  bootserver2_Gateway   2nd boot server gateway. Default=0.0.0.0

    string Bootserver2_Gateway: 2nd boot server gateway. Default=0.0.0.0

  bootserver2_Port   2nd boot server port. Min=1025, Max=65536,
                                Default=6910

    uint Bootserver2_Port: 2nd boot server port. Min=1025, Max=65536,
                    Default=6910

  bootserver3_Ip   3rd boot server IP. Default=0.0.0.0

    string Bootserver3_Ip: 3rd boot server IP. Default=0.0.0.0

  bootserver3_Netmask   3rd boot server netmask. Default=0.0.0.0

    string Bootserver3_Netmask: 3rd boot server netmask. Default=0.0.0.0

  bootserver3_Gateway   3rd boot server gateway. Default=0.0.0.0

    string Bootserver3_Gateway: 3rd boot server gateway. Default=0.0.0.0

```
bootserver3_Port          3rd boot server port. Min=1025, Max=65536,
                          Default=6910

   uint Bootserver3_Port: 3rd boot server port. Min=1025, Max=65536,
                   Default=6910

bootserver4_Ip            4th boot server IP. Default=0.0.0.0

   string Bootserver4_Ip: 4th boot server IP. Default=0.0.0.0

bootserver4_Netmask       4th boot server netmask. Default=0.0.0.0

   string Bootserver4_Netmask: 4th boot server netmask. Default=0.0.0.0

bootserver4_Gateway       4th boot server gateway. Default=0.0.0.0

   string Bootserver4_Gateway: 4th boot server gateway. Default=0.0.0.0

bootserver4_Port          4th boot server port. Min=1025, Max=65536,
                          Default=6910

   uint Bootserver4_Port: 4th boot server port. Min=1025, Max=65536,
                   Default=6910

verboseMode               Display verbose diagnostic information when
                          set to 1. Default=0

   bool VerboseMode: Display verbose diagnostic information when set to
                   true. Default=false

interruptSafeMode         Interrupt safe mode (use if target device
                          hangs during boot) when set to 1. Default=0

   bool InterruptSafeMode: Interrupt safe mode (use if target device
                   hangs during boot) when set to true. Default=false

paeMode                   PAE mode (use if PAE enabled in boot.ini of
                          target device) when set to 1. Default=0

   bool PaeMode: PAE mode (use if PAE enabled in boot.ini of target
                   device) when set to true. Default=false

bootFromHdOnFail          For network recovery reboot to hard drive
                          when set to 1, restore network connection
                          when set to 0. Default=0

   bool BootFromHdOnFail: For network recovery reboot to hard drive when
                   set to true, restore network connection when set to
                   false. Default=false

recoveryTime              When bootFromHdOnFail is 1, this is the
                          number of seconds to wait before reboot to
                          hard drive. Min=10, Max=60000, Default=50

   uint RecoveryTime: When bootFromHdOnFail is 1, this is the number of
                   seconds to wait before reboot to hard drive. Min=10,
                   Max=60000, Default=50

pollingTimeout            Login polling timeout in milliseconds.
                          Min=1000, Max=60000, Default=5000

   uint PollingTimeout: Login polling timeout in milliseconds. Min=1000,
                   Max=60000, Default=5000

generalTimeout            Login general timeout in milliseconds.
                          Min=1000, Max=60000, Default=5000
```

McIi-Set ServerStore

Set the directory path and cache paths of a Server for a Store.

Set-PvsServerStore

Set ServerStore(s) changed values from PvsServerStore object(s), or set one
or more field values for a PvsServerStore.

-p   Parameters needed to identify the record to
                                be changed.

When ServerStore is not passed the parameters below are used:

One of these required

serverId   GUID of a Server that uses the path to get
                                to the Store.

Guid Guid or ServerId: GUID of a Server that uses the path to get to
                the Store.

serverName   Name of a Server that uses the path to get
                                to the Store.

string Name or ServerName: Name of a Server that uses the path to get
                to the Store.

One of these required

storeId   GUID of the Store.

Guid StoreId: GUID of the Store.

storeName   Name of the Store.

string StoreName: Name of the Store.

Required

PvsServerStore[] ServerStore: Array of PvsServerStore objects with
                changed property value(s) to be set. The object(s)
                can come from a pileline.

Optional

SwitchParameter PassThru: If -PassThru is specified, the resulting
                PvsServerStore object(s) are returned.

SwitchParameter Confirm: The impact of this operation is "low". If -
                Confirm is specified, the operation will be
                confirmed. $ConfirmPreference can be set to "low" to
                have confirmation without the Confirm parameter.

-r   Record containing the new values for fields
                                to be changed.

Fields that can have value set

path   Directory path that the Server uses to
                                access the Store. Default="" Max Length=255

string Path: Directory path that the Server uses to access the Store.
                Default="" Max Length=255

```
              cachePath                    Cache path(s) that the Server uses with the
                                           Store. It is an ordered comma-delimited
                                           list. If none are specified the caches will
                                           be placed in the Store cachePath.
                                           Default=None

         string[] CachePath: Cache path(s) that the Server uses with the
                        Store. If none are specified the caches will be
                        placed in the Store cachePath. Default=None
```

Mcli-Set Site

```
   Set one or more field values for a Site.
```

Set-PvsSite

```
   Set Site(s) changed values from PvsSite object(s), or set one or more field
     values for a PvsSite.

     -p                           Parameters needed to identify the record to
                                  be changed.

       When Site is not passed the parameters below are used:

       One of these required

        siteId                    GUID of the Site to Set.

          Guid Guid or SiteId: GUID of the Site to Set.

        siteName                  Name of the Site to Set.

          string Name or SiteName: Name of the Site to Set.

        Required

          PvsSite[] Site: Array of PvsSite objects with changed property
                        value(s) to be set. The object(s) can come from a
                        pileline.

        Optional

          SwitchParameter PassThru: If -PassThru is specified, the resulting
                        PvsSite object(s) are returned.

          SwitchParameter Confirm: The impact of this operation is "low". If -
                        Confirm is specified, the operation will be
                        confirmed. $ConfirmPreference can be set to "low" to
                        have confirmation without the Confirm parameter.

     -r                           Record containing the new values for fields
                                  to be changed.

       Fields that can have value set

        siteName                  Name of the Site. Max Length=50

          string NewName: Name of the Site. Max Length=50

        description               User description. Default="" Max Length=250

          string Description: User description. Default="" Max Length=250

        defaultCollectionId       GUID of the Collection to place new Devices
                                  into automatically. Not used with
                                  defaultCollectionName. Default=null
                                  Length=36
```

Guid DefaultCollectionId: GUID of the Collection to place new Devices into automatically. Not used with defaultCollectionName. Default=00000000-0000-0000-0000-000000000000

defaultCollectionName   Name of the Collection to place new Devices into automatically. Not used with defaultCollectionId. Default=null

string DefaultCollectionName: Name of the Collection to place new Devices into automatically. Not used with DefaultCollectionId. Default=""

inventoryFilePollingInterval The number of seconds between polls for Disk changes in the Stores. Min=1, Max=600, Default=60

uint InventoryFilePollingInterval: The number of seconds between polls for Disk changes in the Stores. Min=1, Max=600, Default=60

enableDiskUpdate   1 when Disk Updated is enabled for the Site, 0 otherwise. Default=0

bool EnableDiskUpdate: True when Disk Updated is enabled for the Site, false otherwise. Default=false

diskUpdateServerId   GUID of the Disk Update Server for the Site. Not used with diskUpdateServerName. Default=null Length=36

Guid DiskUpdateServerId: GUID of the Disk Update Server for the Site. Not used with DiskUpdateServerName. Default=00000000-0000-0000-0000-000000000000

diskUpdateServerName   Name of the Disk Update Server for the Site. Not used with diskUpdateServerId. Default=null

string DiskUpdateServerName: Name of the Disk Update Server for the Site. Not used with DiskUpdateServerId. Default=""

makUser   User name used for MAK activation. Default=null Max Length=64

string MakUser: User name used for MAK activation. Default="" Max Length=64

makPassword   User password used for MAK activation. Default=null Max Length=64

string MakPassword: User password used for MAK activation. Default="" Max Length=64

enableXsProxy   Enable XenServerProxy when set to 1 Default=null

string EnableXsProxy: Enable XenServerProxy when set to 1 Default=""

virtualHostingPoolId   GUID of the VirtualHostingPool object.

Guid VirtualHostingPoolId: GUID of the VirtualHostingPool object.

virtualHostingPoolName   Name of the VirtualHostingPool object.

string VirtualHostingPoolName: Name of the VirtualHostingPool object.

```
        xsPvsSiteUuid                GUID of the XenServer PVS Site.
```

Mcli-Set SiteView

```
   Set one or more field values for a Site View.
```

Set-PvsSiteView

```
     -p                              Parameters needed to identify the record to
                                     be changed.
```

```
      This required
       siteViewId                   GUID of the Site View to Set.
```

```
      or this required & resolution
       siteViewName                 Name of the Site View to Set.
```

```
      One of these resolutions when needed
       siteId                       GUID of the Site.
```

```
       siteName                     Name of the Site.
```

```
     -r                              Record containing the new values for fields
                                     to be changed.
```

```
       siteViewName                 Name of the Site View. Max Length=50
```

```
       description                  User description. Default="" Max Length=250
```

Mcli-Set Store

Set one or more field values for a Store.

Set-PvsStore

Set Store(s) changed values from PvsStore object(s), or set one or more field values for a PvsStore.

    -p   Parameters needed to identify the record to be changed.

When Store is not passed the parameters below are used:

One of these required

  storeId   GUID of the Store to Set.

   Guid Guid or StoreId: GUID of the Store to Set.

  storeName   Name of the Store to Set.

   string Name or StoreName: Name of the Store to Set.

  Required

   PvsStore[] Store: Array of PvsStore objects with changed property value(s) to be set. The object(s) can come from a pileline.

  Optional

   SwitchParameter PassThru: If -PassThru is specified, the resulting PvsStore object(s) are returned.

   SwitchParameter Confirm: The impact of this operation is "low". If -Confirm is specified, the operation will be confirmed. $ConfirmPreference can be set to "low" to have confirmation without the Confirm parameter.

    -r   Record containing the new values for fields to be changed.

  Fields that can have value set

  storeName   Name of the Store. Max Length=50

   string NewName: Name of the Store. Max Length=50

  siteId   GUID of the Site where Administrators of that Site can change this Store. Not used for Farm Stores. siteName can be used instead. Default=null Length=36

   Guid SiteId: GUID of the Site where Administrators of that Site can change this Store. Not used for Farm Stores. SiteName can be used instead. Default=00000000-0000-0000-0000-000000000000

  siteName   Name of the Site where Administrators of that Site can change this Store. Not used for Farm Stores. siteId can be used instead. Default=null

   string SiteName: Name of the Site where Administrators of that Site can change this Store. Not used for Farm Stores. SiteId can be used instead. Default=""

```
description               User description. Default="" Max Length=250
```

string Description: User description. Default="" Max Length=250

```
path                      Default directory path that the Servers use
                          to access this Store. Max Length=255
```

string Path: Default directory path that the Servers use to access
this Store. Max Length=255

```
cachePath                 Default Cache path(s) that the Servers use
                          with this Store. It is an ordered comma-
                          delimited list. If none are specified the
                          caches will be placed in the WriteCache
                          subdirectory of the Store path. Default=None
```

string[] CachePath: Default Cache path(s) that the Servers use with
this Store. If none are specified the caches will be
placed in the WriteCache subdirectory of the Store
path. Default=None

Mcli-Set UpdateTask

```
Set one or more field values for an Update Task.
```

Set-PvsUpdateTask

Set Update Task(s) changed values from PvsUpdateTask object(s), or set one
or more field values for a PvsUpdateTask.

```
-p                        Parameters needed to identify the record to
                          be changed.
```

When UpdateTask is not passed the parameters below are used:

```
This required
```

```
updateTaskId              GUID of the Update Task to Set.
```

Guid Guid or UpdateTaskId: GUID of the Update Task to Set.

```
or this required & resolution
```

```
updateTaskName            Name of the Update Task to Set.
```

string Name or UpdateTaskName: Name of the Update Task to Set.

```
One of these resolutions when needed
```

```
siteId                    GUID of the Site.
```

Guid SiteId: GUID of the Site.

```
siteName                  Name of the Site.
```

string SiteName: Name of the Site.

Required

PvsUpdateTask[] UpdateTask: Array of PvsUpdateTask objects with
changed property value(s) to be set. The object(s)
can come from a pileline.

Optional

SwitchParameter PassThru: If -PassThru is specified, the resulting
PvsUpdateTask object(s) are returned.

-r   Record containing the new values for fields
                                to be changed.

updateTaskName   Name of the Update Task. It is unique within
                                the Site. Max Length=50

description   User description. Default="" Max Length=250

enabled   1 when it will be processed, 0 otherwise.
                                Default=1

hour   The hour of the day to perform the task.
                                Min=0, Max=23, Default=0

minute   The minute of the hour to perform the task.
                                Min=0, Max=59, Default=0

recurrence   The update will reoccur on this schedule. 0
                                = None, 1 = Daily, 2 = Every Weekday, 3 =
                                Weekly, 4 = Monthly Date, 5 = Monthly Type.
                                Min=0, Max=5, Default=0

dayMask   Days selected values. 1 = Monday, 2 =
                                Tuesday, 4 = Wednesday, 8 = Thursday, 16 =
                                Friday, 32 = Saturday, 64 = Sunday, 128 =
                                Day. Default=0. This is used with Weekly and
                                Monthly Type recurrence. Min=1, Max=255,
                                Default=4

date   Comma delimited days of the month. Numbers
                                from 1-31 are the only valid values. This is
                                used with Monthly Date recurrence.
                                Default="" Max Length=83

monthlyOffset   When to happen monthly. 0 = None, 1 = First,
                                2 = Second, 3 = Third, 4 = Forth, 5 = Last.
                                This is used with Monthly Type recurrence.
                                Min=0, Max=5, Default=3

esdType   Esd to use. Valid values are SCCM or WSUS.
                                If no value, a custom script is run on the
                                client. Default=null Max Length=50

preUpdateScript   Script file to run before the update starts.
                                Default=null Max Length=255

preVmScript   Script file to run before the VM is loaded.
                                Default=null Max Length=255

postUpdateScript   Script file to run after the update
                                finishes. Default=null Max Length=255

postVmScript   Script file to run after the VM is unloaded.
                                Default=null Max Length=255

domain   Domain to add the Disk Update Device(s) to.
                                If not included, the first Domain Controller
                                found on the Server is used. Default=null
                                Max Length=255

organizationUnit   Organizational Unit to add the Disk Update
                                Device(s) to. This parameter is optional. If
                                it is not specified, the device is added to
                                the built in Computers container. Child OU's
                                should be delimited with forward slashes,
                                e.g. "ParentOU/ChildOU". Special characters
                                in an OU name, such as '"', '#', '+', ',',
                                ';', '>', '=', must be escaped with a
                                backslash. For example, an OU called

>   "commaIn,TheMiddle" must be specified as
>   "commaIn\,TheMiddle". The old syntax of
>   delimiting child OU's with a comma is still
>   supported, but deprecated. Note that in this
>   case, the child OU comes first, e.g.
>   "ChildOU,ParentOU". Default=null Max
>   Length=255

>   string OrganizationUnit: Organizational Unit to add the Disk Update
>   Device(s) to. This parameter is optional. If it is
>   not specified, the device is added to the built in
>   Computers container. Child OU's should be delimited
>   with forward slashes, e.g. "ParentOU/ChildOU".
>   Special characters in an OU name, such as '"', '#',
>   '+', ',', ';', '>', '=', must be escaped with a
>   backslash. For example, an OU called
>   "commaIn,TheMiddle" must be specified as
>   "commaIn\,TheMiddle". The old syntax of delimiting
>   child OU's with a comma is still supported, but
>   deprecated. Note that in this case, the child OU
>   comes first, e.g. "ChildOU,ParentOU". Default="" Max
>   Length=255

>   postUpdateApprove   Access to place the version in after the
>   update has occurred. 0 = Production, 1 =
>   Test, 2 = Maintenance. Min=0, Max=2,
>   Default=0

>   uint PostUpdateApprove: Access to place the version in after the
>   update has occurred. 0 = Production, 1 = Test, 2 =
>   Maintenance. Min=0, Max=2, Default=0

Mcli-Set VirtualHostingPool

Set one or more field values for a Virtual Hosting Pool.

Set-PvsVirtualHostingPool

Set Virtual Hosting Pool(s) changed values from PvsVirtualHostingPool
   object(s), or set one or more field values for a PvsVirtualHostingPool.

  -p   Parameters needed to identify the record to
                            be changed.

   When VirtualHostingPool is not passed the parameters below are used:

  This required

   virtualHostingPoolId   GUID of the Virtual Hosting Pool to Set.

     Guid Guid or VirtualHostingPoolId: GUID of the Virtual Hosting Pool
                     to Set.

  or this required & resolution

   virtualHostingPoolName   Name of the Virtual Hosting Pool to Set.

     string Name or VirtualHostingPoolName: Name of the Virtual Hosting
                     Pool to Set.

  One of these resolutions when needed

   siteId   GUID of the Site.

    Guid SiteId: GUID of the Site.

siteName   Name of the Site.

  string SiteName: Name of the Site.

Required

  PvsVirtualHostingPool[] VirtualHostingPool: Array of
                PvsVirtualHostingPool objects with changed property
                value(s) to be set. The object(s) can come from a
                pileline.

Optional

  SwitchParameter PassThru: If -PassThru is specified, the resulting
                PvsVirtualHostingPool object(s) are returned.

  SwitchParameter Confirm: The impact of this operation is "low". If -
                Confirm is specified, the operation will be
                confirmed. $ConfirmPreference can be set to "low" to
                have confirmation without the Confirm parameter.

-r   Record containing the new values for fields
                to be changed.

 Fields that can have value set

virtualHostingPoolName   Name of the Virtual Hosting Pool. It is
                unique within the Site. Max Length=50

  string NewName: Name of the Virtual Hosting Pool. It is unique within
                the Site. Max Length=50

type   Type of the Virtual Hosting Pool. 0 = Citrix
                XenServer, 1 = Microsoft SCVMM/Hyper-V, 2 =
                VMWare vSphere/ESX. Min=0, Max=3, Default=0

  uint Type: Type of the Virtual Hosting Pool. 0 = Citrix XenServer, 1
                = Microsoft SCVMM/Hyper-V, 2 = VMWare vSphere/ESX.
                Min=0, Max=3, Default=0

description   User description. Default="" Max Length=250

  string Description: User description. Default="" Max Length=250

server   Name or IP of the Host Server. Max
                Length=255

  string Server: Name or IP of the Host Server. Max Length=255

port   Port of the Host Server. Min=80, Max=65534,
                Default=80

  uint Port: Port of the Host Server. Min=80, Max=65534, Default=80

datacenter   Datacenter of the Virtual Hosting Pool.
                Default="" Max Length=250

  string Datacenter: Datacenter of the Virtual Hosting Pool. Default=""
                Max Length=250

updateLimit   Number of updates at the same time. Min=2,
                Max=1000, Default=1000

  uint UpdateLimit: Number of updates at the same time. Min=2,
                Max=1000, Default=1000

updateTimeout   Timeout for updates. Min=2, Max=240,
                         Default=60

 uint UpdateTimeout: Timeout for updates. Min=2, Max=240, Default=60

shutdownTimeout   Timeout for shutdown. Min=2, Max=30,
                         Default=10

 uint ShutdownTimeout: Timeout for shutdown. Min=2, Max=30, Default=10

userName   Name to use when logging into the Server.

 string UserName: Name to use when logging into the Server.

password   Password to use when logging into the
                         Server.

 string Password: Password to use when logging into the Server.

xdHostingUnitUuid   UUID of XenDesktop Hosting Unit Default=null
                         Length=36

 Guid XdHostingUnitUuid: UUID of XenDesktop Hosting Unit
                  Default=00000000-0000-0000-0000-000000000000

prepopulateEnabled   Enable prepopulate when set to 1 Default=0

 bool PrepopulateEnabled: Enable prepopulate when set to true
                  Default=false

xsPvsSiteUuid   UUID of XenServer PVS_site Default=null
                         Length=36

 Guid XsPvsSiteUuid: UUID of XenServer PVS_site Default=00000000-
                  0000-0000-0000-000000000000

platformVersion   Hypervisor Host Version Default=null Max
                         Length=250

 string PlatformVersion: Hypervisor Host Version Default="" Max
                  Length=250

xdHcHypervisorConnectionName Hypervisor Connection Name for HCL
                         Connection Details object Default=null Max
                         Length=250

 string XdHcHypervisorConnectionName: Hypervisor Connection Name for
                  HCL Connection Details object Default="" Max
                  Length=250

xdHcHypervisorConnectionUid Hypervisor Connection Uid for HCL
                         Connection Details object Default=null Max
                         Length=250

 string XdHcHypervisorConnectionUid: Hypervisor Connection Uid for HCL
                  Connection Details object Default="" Max Length=250

xdHcRevision   Revision for HCL Connection Details object
                         Default=null Max Length=250

 string XdHcRevision: Revision for HCL Connection Details object
                  Default="" Max Length=250

xdHcCustomProperties   Custom Properties for HCL Connection Details
                         object Default=null Max Length=250

 string XdHcCustomProperties: Custom Properties for HCL Connection
                  Details object Default="" Max Length=250

```
           xdHcSslThumbprints            Ssl Thumbprints for HCL Connection Details
                                          object Default=null Max Length=250

              string XdHcSslThumbprints: Ssl Thumbprints for HCL Connection Details
                                 object Default="" Max Length=250

           disableHostXsProxy            True to disable PVS-Accelerator Default=null

              string DisableHostXsProxy: True to disable PVS-Accelerator Default=""
```

Mcli-Set XDSite

Set the configServices field value for a XenDesktop Site.

Set-PvsXDSite

Set XDSite(s) changed values from PvsXDSite object(s), or set one or more
   field values for a PvsXDSite.

     -p   Parameters needed to identify the record to
                                    be changed.

       When XDSite is not passed the parameters below are used:

       This required

         xdSiteId   GUID of the XenDesktop Site to Set.

           Guid Guid or XdSiteId: GUID of the XenDesktop Site to Set.

         Required

           PvsXDSite[] XDSite: Array of PvsXDSite objects with changed property
                       value(s) to be set. The object(s) can come from a
                       pileline.

         Optional

           SwitchParameter PassThru: If -PassThru is specified, the resulting
                       PvsXDSite object(s) are returned.

           SwitchParameter Confirm: The impact of this operation is "low". If -
                       Confirm is specified, the operation will be
                       confirmed. $ConfirmPreference can be set to "low" to
                       have confirmation without the Confirm parameter.

     -r   Record containing the new values for fields
                                    to be changed.

       Fields that can have value set

         configServices   Comma delimited list of XenDesktop Server
                                    addresses. Max Length=2000

           string[] ConfigServices: XenDesktop Server addresses. Max Length=2000

Mcli-SetList

```
Mcli-SetList commandName[ -p name=value[, name2=value2]][ -r
  record1field=value[, record1field2=value2][, record2field=value[,
  record2field2=value2]]]
```

Mcli-SetList DeviceBootstraps

Set the Bootstrap Files for a Device.

Set-PvsDeviceBootstrap

Set Device Bootstrap List(s) changed values from PvsDeviceBootstrap object(s).

  -p   Parameters needed to identify the record to
                                be changed.

  One of these required

   deviceId   GUID of the Device.

   deviceName   Name of the Device.

   deviceMac   MAC of the Device.

   Required

     PvsDeviceBootstrap[] DeviceBootstrap: Array of PvsDeviceBootstrap
                       objects with changed DeviceBootstrap. The object(s)
                       can come from a pileline.

   Optional

     SwitchParameter PassThru: If -PassThru is specified, the resulting
                       PvsDeviceBootstrap object(s) are returned.

     SwitchParameter Confirm: The impact of this operation is "low". If -
                       Confirm is specified, the operation will be
                       confirmed. $ConfirmPreference can be set to "low" to
                       have confirmation without the Confirm parameter.

  -r   Records with fields to set.

   These exist in the DeviceBootstrap array within each
                   PvsDeviceBootstrap.

   Each array item is a PvsDeviceBootstrapList object.

   They are set using the Add, Insert, Remove, Set, and Reorder methods in
                   the PvsDeviceBootstrap.

   bootstrap   Name of the bootstrap file. Max Length=259

     string Name or Bootstrap: Name of the bootstrap file. Max Length=259

   menuText   Text that is displayed in the Boot Menu. If
                              this field has no value, the bootstrap value
                              is used. Default="" ASCII Max Length=64

     string MenuText: Text that is displayed in the Boot Menu. If this
                       field has no value, the bootstrap value is used.
                       Default="" ASCII Max Length=64

Mcli-SetList DevicePersonality

Set the Device Personality names and values.

Set-PvsDevicePersonality

Set Device Personality List(s) changed values from PvsDevicePersonality object(s).

  -p   Parameters needed to identify the record to
                                be changed.

  One of these required

```
     deviceId                    GUID of the Device.

     deviceName                  Name of the Device.

     deviceMac                   MAC of the Device.
```

Required

  PvsDevicePersonality[] DevicePersonality: Array of
                 PvsDevicePersonality objects with changed
                 DevicePersonality. The object(s) can come from a
                 pileline.

Optional

  SwitchParameter PassThru: If -PassThru is specified, the resulting
                 PvsDevicePersonality object(s) are returned.

  SwitchParameter Confirm: The impact of this operation is "low". If -
                 Confirm is specified, the operation will be
                 confirmed. $ConfirmPreference can be set to "low" to
                 have confirmation without the Confirm parameter.

```
-r                            Records with fields to set.
```

  These exist in the DevicePersonality array within each
                 PvsDevicePersonality.

  Each array item is a PvsDevicePersonalityList object.

  They are set using the Add, Insert, Remove, Set, and Reorder methods in
                 the PvsDevicePersonality.

```
   name                        Name of the Device personality item. Max
                                 Length=250
```

   string Name: Name of the Device personality item. Max Length=250

```
   value                       Value for the Device personality item. Max
                                 Length=1000
```

   string Value: Value for the Device personality item. Max Length=1000