

vSphere 6.0 Webcast Series

PowerCLI 6.0 R1

Brian Graf
Senior Technical Marketing Manager

vmware®

© 2014 VMware Inc. All rights reserved.

Microsoft® PowerShell™ - Overview

- Microsoft PowerShell is the foundation of PowerCLI
- The best shell today for Windows Environments
 - Easy to learn and understand
 - Built for **system administrators**, not for programmers
 - Object Orientated
 - Extended to the entire Datacenter
- Microsoft is serious about it
So is VMware



PowerCLI - Overview

- Commandline tool that leverages Microsoft PowerShell
- The best shell for automating and managing your virtual environment
 - Follows PowerShell syntax and command verbs
 - Built for **administrators**, not for programmers
 - Almost 500 cmdlets for working with vSphere environments
 - 10+ Products and technologies, and more to come!
 - PowerCLI 6.0 is backwards compatible to vSphere 5.0
 - No license required



PowerCLI 6.0 Release 1 – What's New

- Modules/Snapin Hybrid Mode
- .Net 4.5 adoption
- Guest Cmdlet deprecation
- vCloud Air connection Cmdlets
- vCloud Air Management
- vCloud Suite SDK Cmdlets
- VSAN Cmdlets
- DRM Dump Info Cmdlet

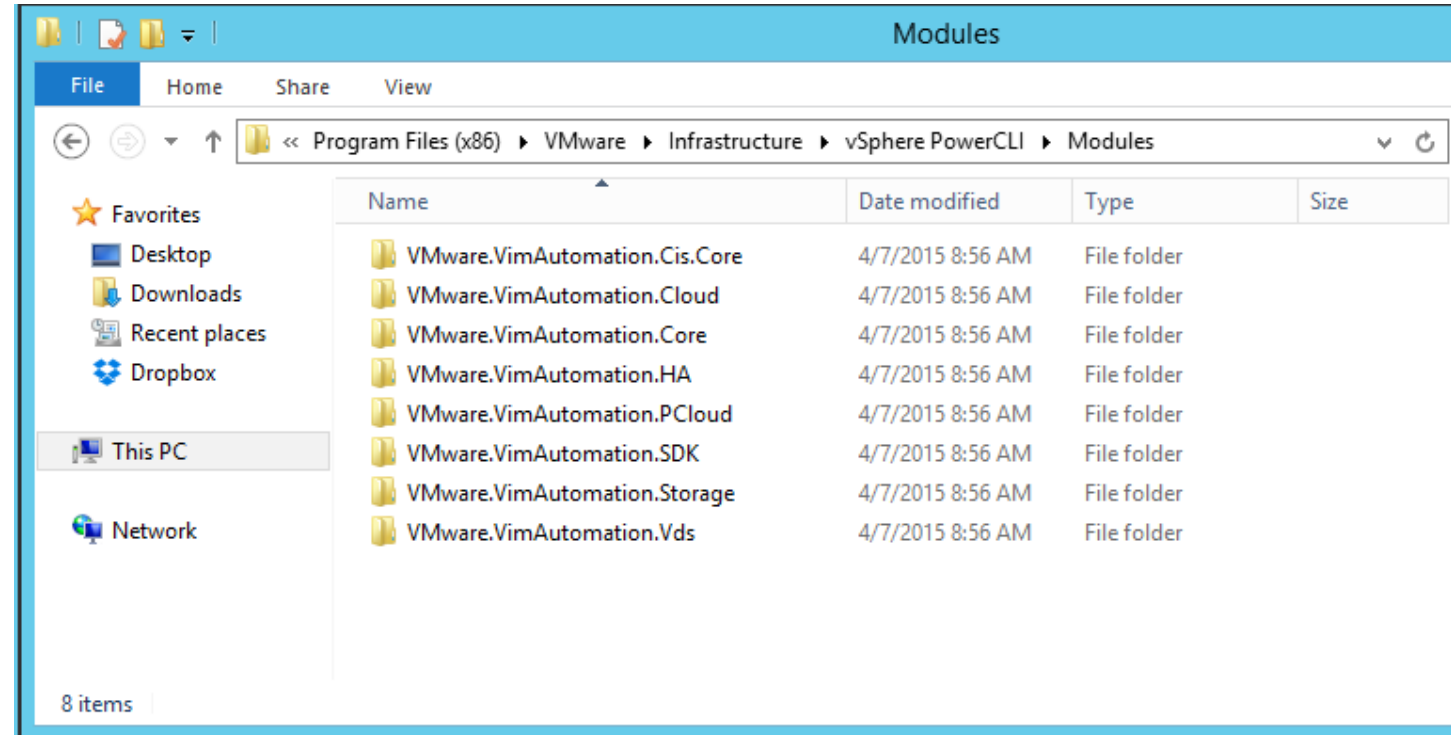


PowerCLI 6.0 Release 1

Core Enhancements

New in PowerCLI 6.0: Modules/Snapin Hybrid Mode

- Moving toward PowerCLI Modules
- 6.0 is hybrid (Snapin and Modules)
- Snapins require an installer
 - Registry Entries
- Modules use 'Import-Module' cmdlet
- Module can consist of additional
 - Scripts
 - Manifests
 - Binaries
 - and more
- Future: Install-Module PowerCLI



New in PowerCLI 6.0: .Net 4.5 Adoption

- Moved away from .Net 2.0 and PowerShell 2.0
- Added benefits of .Net 4.5
 - PowerShell V4
 - Windows 8.1 and Windows Server 2012 now supported



Guest Cmdlet Deprecation

- Leveraging alternatives allowing more stability and community involvement
 - Invoke-VMScript <cmdline settings>

Deprecated Cmdlets

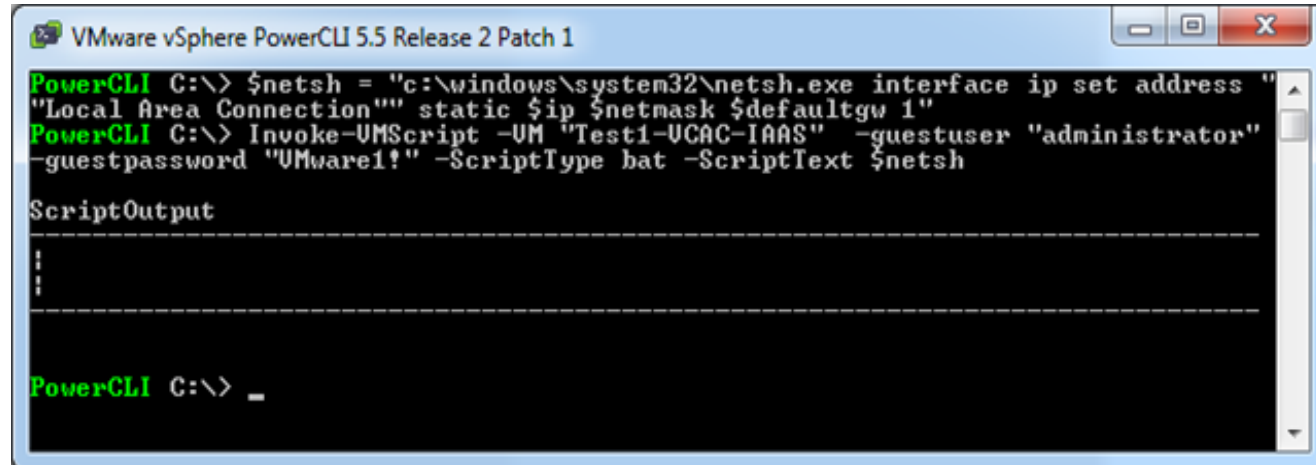
Get-VMGuestNetworkInterface

Set-VMGuestNetworkInterface

Get-VMGuestRoute

New-VMGuestRoute

Remove-VMGuestRoute

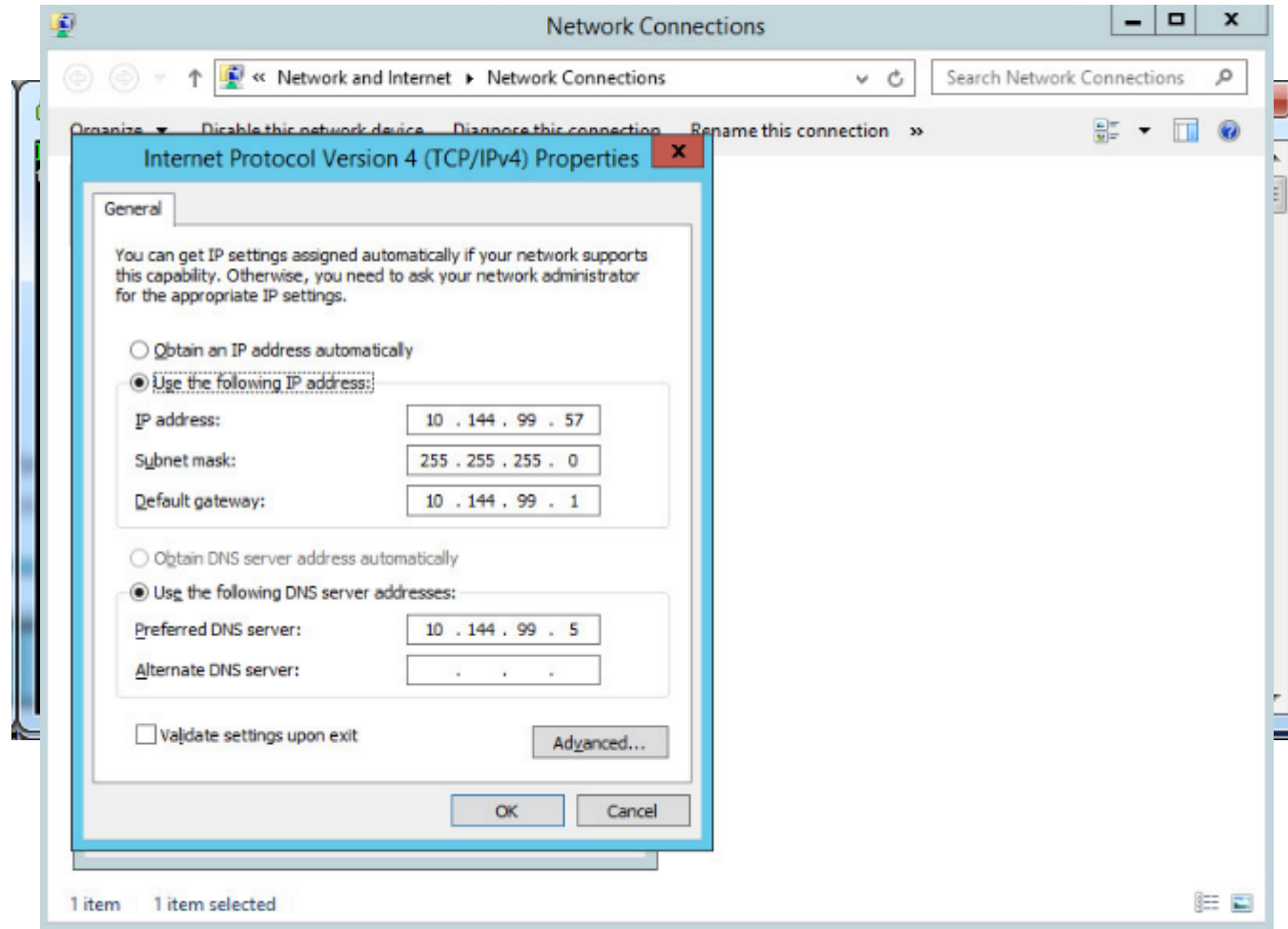


```
VMware vSphere PowerCLI 5.5 Release 2 Patch 1
PowerCLI C:\> $netsh = "c:\windows\system32\netsh.exe interface ip set address "
"Local Area Connection" static $ip $netmask $defaultgw 1"
PowerCLI C:\> Invoke-VMScript -UM "Test1-UCAC-IAAS" -guestuser "administrator"
-guestpassword "VMware1!" -ScriptType bat -ScriptText $netsh

ScriptOutput
-----
:
:
-----

PowerCLI C:\> _
```


Guest Cmdlet Deprecation - Example



New in PowerCLI 6.0: Renamed Cmdlets

Original Cmdlet	New Cmdlet
Shutdown-VMGuest	Stop-VMGuest
Apply-VMHostProfile	Invoke-VMHostProfile
Apply-DrsRecommendation	Invoke-DRSRecommendation

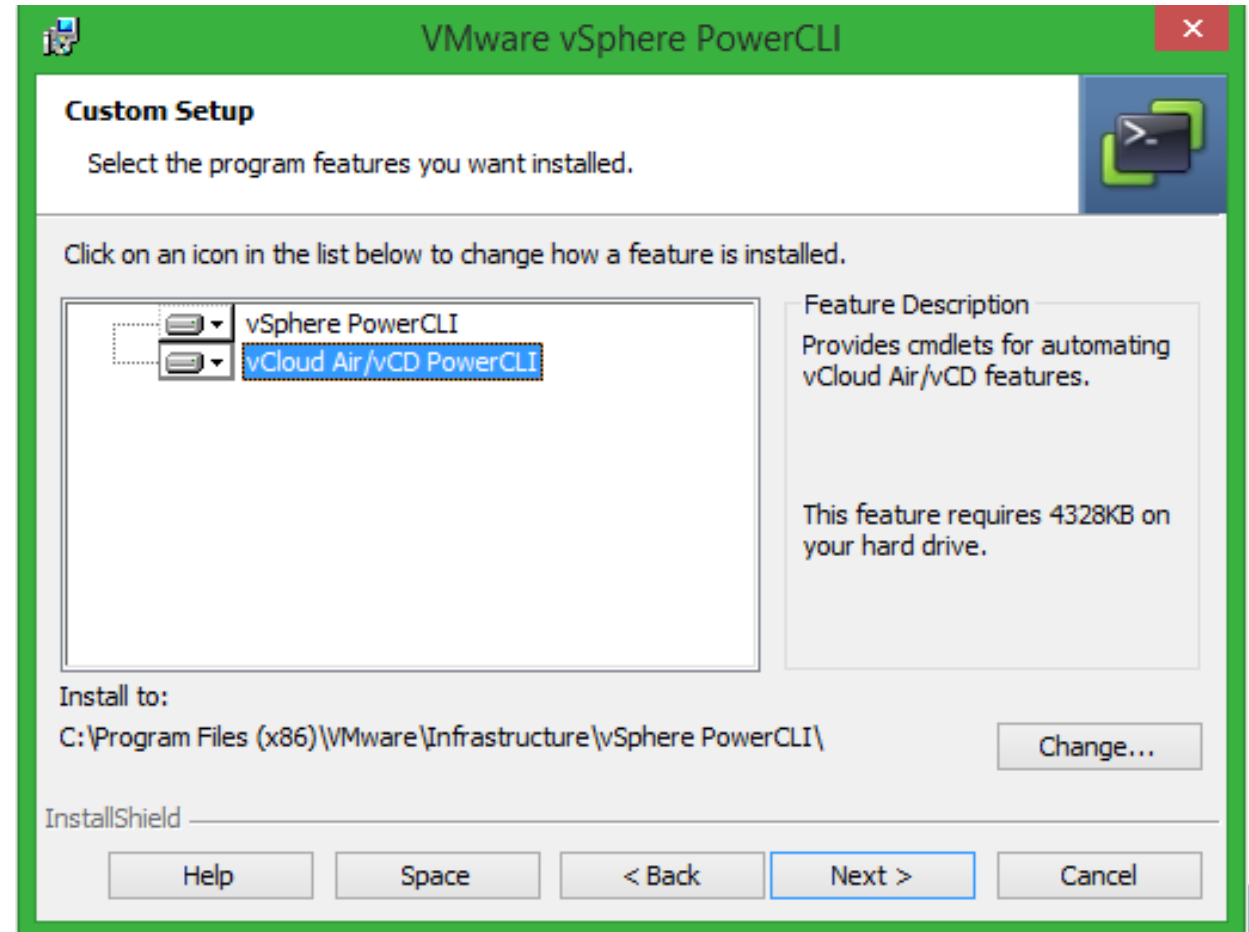
- Cmdlets renamed to be consistent with Microsoft's naming standards.
- Original cmdlet names added on as aliases to not break users scripts.

vCloud Air Cmdlets

New in PowerCLI 6.0: Manage vCloudAir

- Manage On-Premises and vCloud air in the same console
- 100+ Cmdlets to work with the Cloud
- “Tenant” PowerCLI removed
- Optional Install in PowerCLI
- Connect to multiple Clouds
- Get-CIView gives entire public API
- Store credentials for easy connection

- **Report**
- **Troubleshoot**
- **Automate**



New in PowerCLI 6.0: Manage vCloud Air

vCloud Air

New Cmdlets:

```
Connect-PIServer  
Disconnect-PIServer  
Get-PIDatacenter
```

Updated Cmdlets:

“CI” Cmdlets now have a “PI” Alias and work with vCloud Air

Example PowerCLI code for some of the new vCloud Air cmdlets:

Connect to default vCloud Air Instance

```
Connect-PIServer -Username me@mycompany.com -Password VMware1!
```

List all vCloud Air Datacenters you have access to

```
Get-PIDatacenter
```

Connect to all vCloud Air Datacenters you have access to

```
Get-PIDatacenter | Connect-PIDatacenter
```

New in PowerCLI 6.0: Manage vCloud Air

vCloud Air

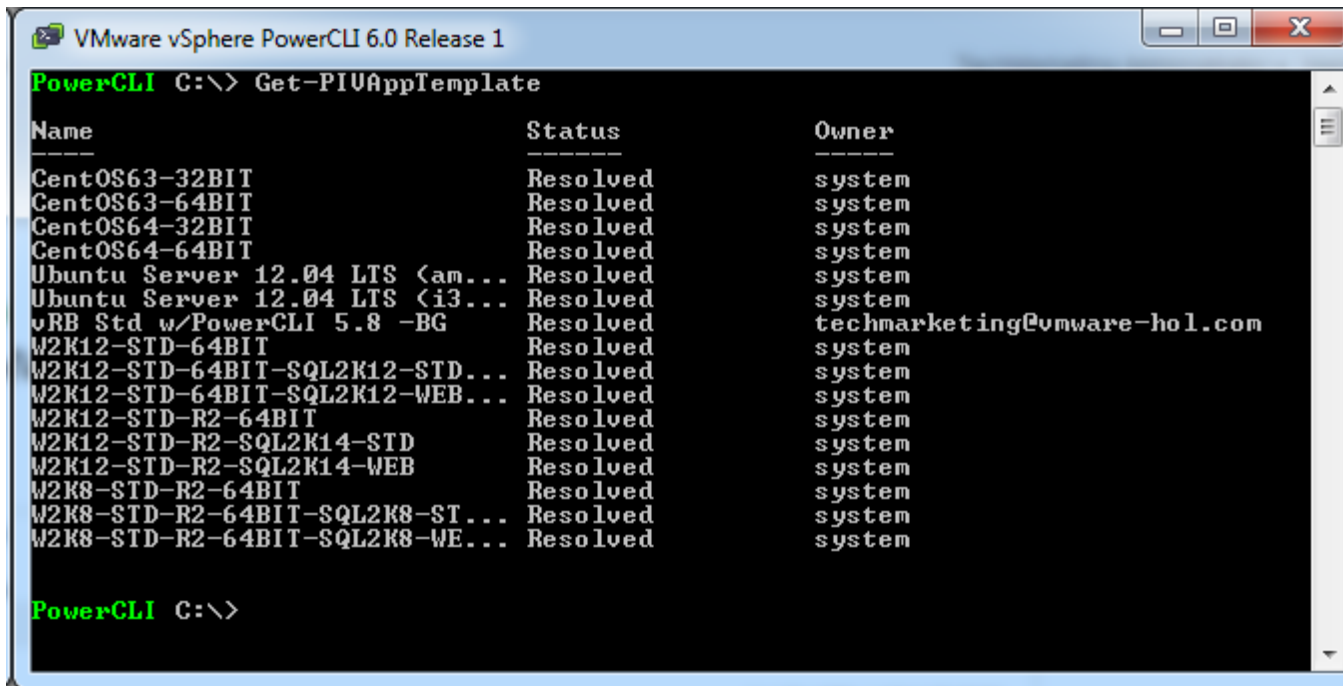
New Cmdlets:

Connect-PIServer
Disconnect-PIServer
Get-PIDatacenter

Updated Cmdlets:

“CI” Cmdlets now have a “PI” Alias and work with vCloud Air

Example PowerCLI code for some of the new vCloud Air cmdlets:



```
VMware vSphere PowerCLI 6.0 Release 1
PowerCLI C:\> Get-PIAppTemplate

Name                               Status      Owner
----                               -
CentOS63-32BIT                     Resolved    system
CentOS63-64BIT                     Resolved    system
CentOS64-32BIT                     Resolved    system
CentOS64-64BIT                     Resolved    system
Ubuntu Server 12.04 LTS (am...     Resolved    system
Ubuntu Server 12.04 LTS (i3...     Resolved    system
vRB Std w/PowerCLI 5.8 -BG         Resolved    techmarketing@vmware-hol.com
W2K12-STD-64BIT                    Resolved    system
W2K12-STD-64BIT-SQL2K12-STD...     Resolved    system
W2K12-STD-64BIT-SQL2K12-WEB...     Resolved    system
W2K12-STD-R2-64BIT                 Resolved    system
W2K12-STD-R2-SQL2K14-STD           Resolved    system
W2K12-STD-R2-SQL2K14-WEB           Resolved    system
W2K8-STD-R2-64BIT                  Resolved    system
W2K8-STD-R2-64BIT-SQL2K8-ST...     Resolved    system
W2K8-STD-R2-64BIT-SQL2K8-WE...     Resolved    system

PowerCLI C:\>
```

Manage vCloud Air Examples

vCloud Air

Example PowerCLI code for some vCloud Air cmdlets:

Retrieve a virtual machine template named "myTemplateVM", create a virtual machine named "myVM" from the template, and add it to a cloud vApp named "myPIVApp".

```
$templateVM = Get-PIVMTemplate -Name "myTemplateVM"  
Get-PIVApp "myPIVApp" | New-PIVM -Name "myVM" -VMTemplate $templateVM
```

List all Edge Gateways

```
Search-Cloud -QueryType EdgeGateway
```

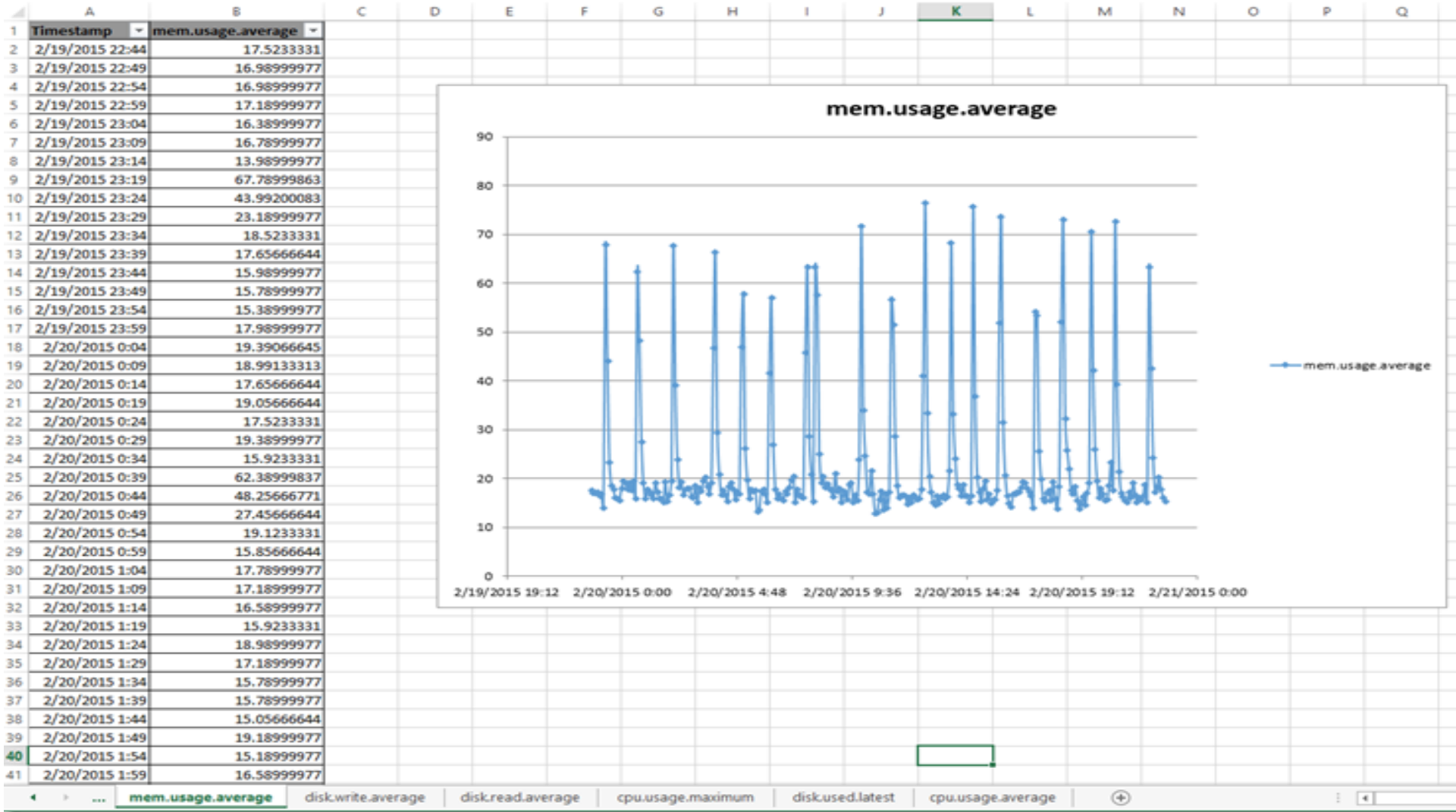
Create a new vApp from a specified vApp template and store it in the specified organization vDC.

```
$myTemplate = Get-PIVAppTemplate -Name 'MySourceTemplate'  
$myOrgVdc = Get-OrgVdc -Name 'MyOrgVdc'  
New-PIVApp -Name 'MyNewVApp' -Description "This vApp is created from the vApp template named  
MySourceTemplate." -OrgVdc $myOrgVdc -VAppTemplate $myTemplate
```

Manage vCloud Air Examples



Easily export information to create complex vCloud Air Reports



vCloud Suite SDK Cmdlets

New in PowerCLI 6.0: vCloud Suite SDK Cmdlets

vCloud Air

New Cmdlets:

```
Connect-CisServer  
Get-CisService
```

Covered:

Low level access to all API (Get-View type access)

- Tagging
- Content Library

Example PowerCLI code for some of the new vCloud Suite SDK cmdlets:

Connect to default vCloud Suite Service Server

```
Connect-CisServer -Username me@mycompany.com -Password vmware1!
```

List all services you can work with on the vCloud Suite Service Server

```
Get-CisService
```

List a particular service to work with

```
Get-CisService -Name 'com.vmware.cis.tagging.tag'
```

New in PowerCLI 6.0: vCloud Suite SDK Cmdlets

vCloud
Suite SDK

```
VMware vSphere PowerCLI 6.0 Release 1
PowerCLI C:\> $ContentLibrary = Get-CisService com.vmware.content.local_library
PowerCLI C:\> $ContentLibrary | Get-Member

TypeName: VMware.VimAutomation.Cis.Core.Impl.U1.CisServiceBase

Name      MemberType Definition
-----
create    CodeMethod id create(string? client_token, com.vmware.content....
delete    CodeMethod void delete(id library_id)
get        CodeMethod com.vmware.content.library_model get(id library_id)
list      CodeMethod List<id> list()
update    CodeMethod void update(id library_id, com.vmware.content.libra...
Equals     Method      bool Equals(System.Object obj)
GetHashCode Method      int GetHashCode()
GetType    Method      type GetType()
ToString   Method      string ToString()
Constants NoteProperty System.Management.Automation.PSCustomObject Constants=
Help       NoteProperty System.Management.Automation.PSCustomObject Help=@<...
Name       Property     string Name {get;}
Uid        Property     string Uid {get;}

PowerCLI C:\>
```

New in PowerCLI 6.0: vCloud Suite SDK Cmdlets

vCloud
Suite SDK

```
VMware vSphere PowerCLI 6.0 Release 1

PowerCLI C:\> $ContentLibrary.list()

Value
-----
fd810c5e-0c6f-4f11-b4b3-3d08e67941e8

PowerCLI C:\> $ContentLibrary.get($ContentLibrary.list().Value)

Help      : @<Documentation=The <@link LibraryModel> <@term structure> represents a
           Content Library resource model. <p> A <@name LibraryModel> is a container for
           a set of items which represent a usable set of files. The Content Library
           Service allows for multiple libraries to be created with separate
           authorization and sharing policies. <p> Each <@name LibraryModel> is a
           container for a set of <@link ItemModel> instances. Each item is a logical
           object in a library, which may have multiple files. <p> A <@name
           LibraryModel> may be local or subscribed. A local library has its source of
           truth about items within this Content Library Service. Items may be added to
           or removed from the library. A local library may also be private or
           published. When published, the library is exposed by a network endpoint and
           can be used by another Content Library Service for synchronization. A private
           local library cannot be used for synchronization. <p> A subscribed library is
           a library which gets its source of truth from another library that may be
           across a network in another Content Library Service. A subscribed library may
           have a different name and metadata from the library to which it subscribes,
           but the set of library items is always the same as those in the source
           library. Library items cannot be manually added to a subscribed library --
           they can only be added by adding new items to the source library.; id=;
           creation_time=; description=; last_modified_time=; last_sync_time=; name=;
           storage_backings=; type=; version=; publish_info=; subscription_info=;
           server_guid=)
id         : fd810c5e-0c6f-4f11-b4b3-3d08e67941e8
publish_info : @<Help=; user_name=vcsp; publish_url=https://172.16.88.200/cls/vcsp/lib/fd810c
           5e-0c6f-4f11-b4b3-3d08e67941e8/lib.json; authentication_method=NONE;
           published=False; password=; persist_json_enabled=False>
storage_backings : @<Help=; datastore_id=datastore-30; storage_uri=; type=DATASTORE>
description  : A New sample Content Library from PowerCLI
last_modified_time : 2/3/2015 3:11:02 PM
name         : New Content Library 2
creation_time  : 2/3/2015 3:11:02 PM
subscription_info :
type         : LOCAL
server_guid   : ad6ce7e6-29c7-4689-92da-fa55b5b52d05
```

New in PowerCLI 6.0: vCloud Suite SDK Cmdlets

vCloud
Suite SDK

```
# Create a local content library on an existing datastore
$createSpec = $ContentLibrary.help.create.create_spec.CreateExample()
$createSpec.server_guid = $null
$createSpec.name = "New Content Library 2"
$createSpec.description = "A New sample Content Library from PowerCLI"
$createSpec.type = "LOCAL"
$createSpec.publish_info.persist_json_enabled = $false
$createSpec.publish_info.published = $false
$datastoreID = [VMware.VimAutomation.Cis.Core.Types.V1.ID]$datastoreID
$StorageSpec = New-Object PSObject -Property @{
    datastore_id = $datastoreID
    type         = "DATASTORE"
}
$CreateSpec.storage_backings.Add($StorageSpec)
$UniqueID = [guid]::NewGuid().ToString()
$ContentLibrary.create($UniqueID, $createSpec)

# Creating a friendly PowerShell Advanced Function
Function Get-ContentLibrary ($Name) {
    $ContentLibrary = Get-CisService com.vmware.content.local_library
    $LibraryIDs = $ContentLibrary.list()
    Foreach ($Library in $LibraryIDs) {
        if ($Name) {
            $ContentLibrary.get($Library.Value) | Where { $_.name -eq $Name } | Select Name, Type, Creation_Time, Last_Modified_
        } else {
            $ContentLibrary.get($Library.Value) | Select Name, Type, Creation_Time, Last_Modified_Time, Storage_Backings
        }
    }
}
```

New in PowerCLI 6.0: vCloud Suite SDK Cmdlets

vCloud
Suite SDK

```
VMware vSphere PowerCLI 6.0 Release 1
PowerCLI C:\> # Creating a friendly PowerShell Advanced Function
>>> Function Get-ContentLibrary <$Name> {
>>>     $ContentLibrary = Get-CisService com.vmware.content.local_library
>>>     $LibraryIDs = $ContentLibrary.list()
>>>     Foreach ($Library in $LibraryIDs) {
>>>         if (<$Name> {
>>>             $ContentLibrary.get($Library.Uvalue) | Where < $_.name -eq $Name > | Select Name, Type,
Creation_Time, Last_Modified_Time, Storage_Backings
>>>         } else {
>>>             $ContentLibrary.get($Library.Uvalue) | Select Name, Type, Creation_Time, Last_Modified_Time, Storage_Backings
>>>         }
>>>     }
>>> }
PowerCLI C:\> Get-ContentLibrary

name           : New Content Library 2
type           : LOCAL
creation_time  : 2/3/2015 3:11:02 PM
last_modified_time : 2/3/2015 3:11:02 PM
storage_backings : {@{Help=; datastore_id=datastore-30; storage_uri=; type=DATASTORE}}
```

VSAN Cmdlets



New in PowerCLI 6.0: VSAN Cmdlets

New Cmdlets:

```
Get-VsanDisk           Get-VsanDiskGroup
New-VsanDisk           New-VsanDiskGroup
Remove-VsanDisk        Remove-VsanDiskGroup
```

Updated Cmdlets:

```
New-Cluster           New-VMHostNetworkAdapter
Set-Cluster           Set-VMHostNetworkAdapter
```

Example PowerCLI code for some of the new VSAN cmdlets:

Enable the VSAN VMKernel Network for a host

```
Get-VMHost Host1 | Get-VMHostNetworkAdapter -VMKernel | Where {$_.PortGroupName -eq "Management Network" } | Set-VMHostNetworkAdapter -VsanTrafficEnabled $true -Confirm:$false
```

Enable VSAN on Cluster1 and set to Automatic Disk Claim Mode

```
Get-Cluster Cluster1 | Set-Cluster -VsanEnabled:$true -VsanDiskClaimMode Automatic -Confirm:$false
```

List Hosts and Disk Groups

```
Get-VsanDiskGroup | Select VMHost, Name | FT -AutoSize
```


New in PowerCLI 6.0: VSAN Cmdlets

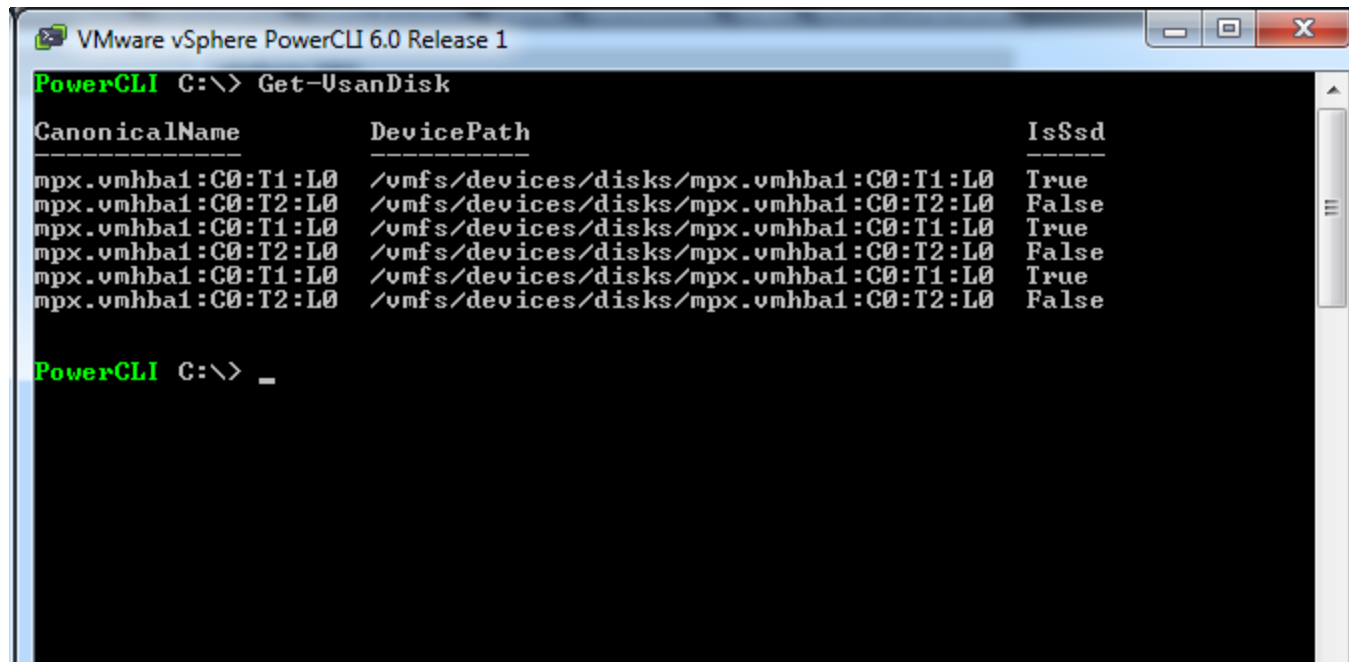
New Cmdlets:

Get-VsanDisk	Get-VsanDiskGroup
New-VsanDisk	New-VsanDiskGroup
Remove-VsanDisk	Remove-VsanDiskGroup

Updated Cmdlets:

New-Cluster	New-VMHostNetworkAdapter
Set-Cluster	Set-VMHostNetworkAdapter

Example PowerCLI code for some of the new VSAN cmdlets:



```
VMware vSphere PowerCLI 6.0 Release 1
PowerCLI C:\> Get-VsanDisk

CanonicalName      DevicePath          IsSsd
-----
mpx.vmhba1:C0:T1:L0 /vmfs/devices/disks/mpx.vmhba1:C0:T1:L0 True
mpx.vmhba1:C0:T2:L0 /vmfs/devices/disks/mpx.vmhba1:C0:T2:L0 False
mpx.vmhba1:C0:T1:L0 /vmfs/devices/disks/mpx.vmhba1:C0:T1:L0 True
mpx.vmhba1:C0:T2:L0 /vmfs/devices/disks/mpx.vmhba1:C0:T2:L0 False
mpx.vmhba1:C0:T1:L0 /vmfs/devices/disks/mpx.vmhba1:C0:T1:L0 True
mpx.vmhba1:C0:T2:L0 /vmfs/devices/disks/mpx.vmhba1:C0:T2:L0 False

PowerCLI C:\> _
```

Distributed Resource Management Information Cmdlets

New in PowerCLI 6.0: Distributed Resource Management Info

Resource
Management

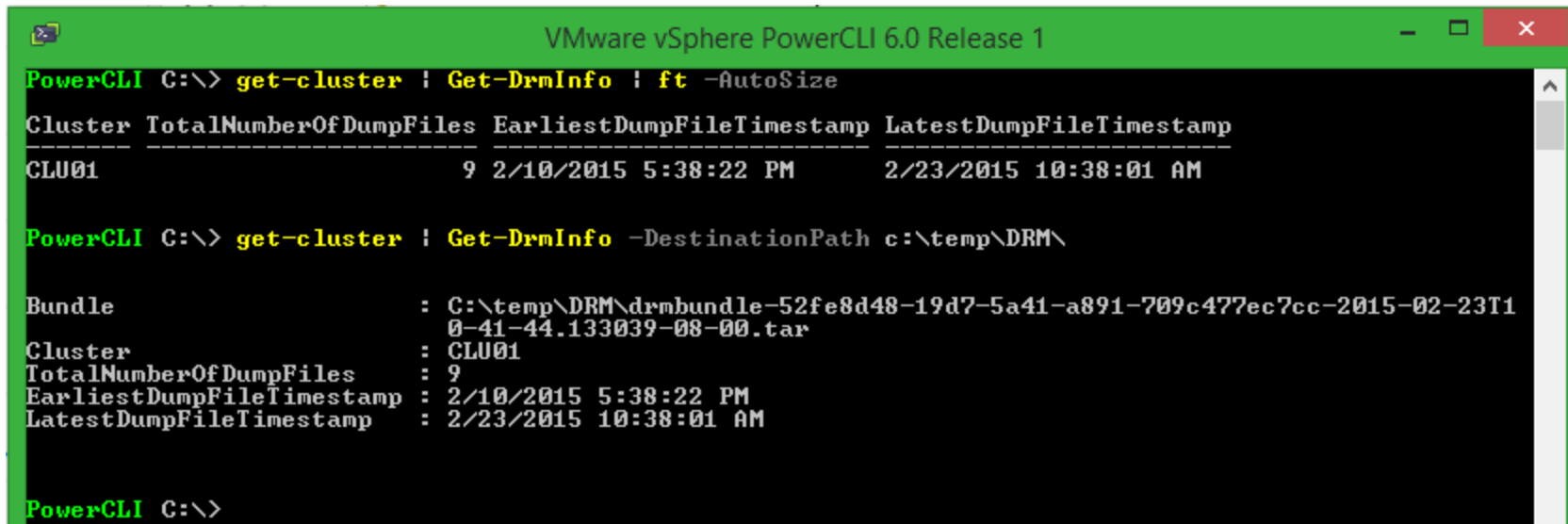
New Cmdlet:

Get-DRMInfo

Example PowerCLI code for some of the new DRMInfo cmdlet:

Downloads all DRM bundles from the Cluster01, that are created in the specified time range, to the specified destination path.

```
Get-DrmInfo -Cluster Cluster01 -Start 2015-01-07T14:15:01 -Finish 2015-01-07T14:15:01 -  
DestinationPath "C:\temp"
```



```
VMware vSphere PowerCLI 6.0 Release 1  
PowerCLI C:\> get-cluster | Get-DrmInfo | ft -AutoSize  
Cluster TotalNumberOfDumpFiles EarliestDumpFileTimestamp LatestDumpFileTimestamp  
-----  
CLU01          9 2/10/2015 5:38:22 PM      2/23/2015 10:38:01 AM  
  
PowerCLI C:\> get-cluster | Get-DrmInfo -DestinationPath c:\temp\DRM\  
  
Bundle          : C:\temp\DRM\drmbundle-52fe8d48-19d7-5a41-a891-709c477ec7cc-2015-02-23T1  
0-41-44.133039-08-00.tar  
Cluster         : CLU01  
TotalNumberOfDumpFiles : 9  
EarliestDumpFileTimestamp : 2/10/2015 5:38:22 PM  
LatestDumpFileTimestamp  : 2/23/2015 10:38:01 AM  
  
PowerCLI C:\>
```

What Now?:

7 Question PowerCLI Survey: <http://bit.ly/powerclisurvey>

Download PowerCLI 6.0 R1 Now!: <http://bit.ly/PowerCLI6r1>

Download PowerCLI reference poster (PDF): <http://bit.ly/PowerCLIPosterv2>

Learn More:

Twitter: @PowerCLI

Blog: <http://blogs.vmware.com/PowerCLI>

Facebook: <https://www.facebook.com/vmwarepowercli>

LinkedIn: <https://www.linkedin.com/groups/PowerCLI-Users-162324/about>