VMware vSphere Pro Series

TRAINSIGNAL

THE GLOBAL LEADER IN PROFESSIONAL COMPUTER TRAINING™

TRAINSIGNAL

THE GLOBAL LEADER IN PROFESSIONAL COMPUTER TRAINING™

Reporting with PowerCLI

VMware vSphere Pro Series

Instructor: Hal Rottenberg, vExpert

In This Lesson:

Creating Great Reports

- ❑How to Create Great Reports - A Brief Overview
  - Who Is Your Audience?
  - What Are Your Requirements?
    - Output Formats
    - Report Destinations
  - Interpret the Data
  - Automate! If You Repeat It, Script It!
- ❑When Is PowerCLI Useful for Reporting?
- ❑Using PowerCLI to Create Reports

In This Lesson:

Actual Scripts!

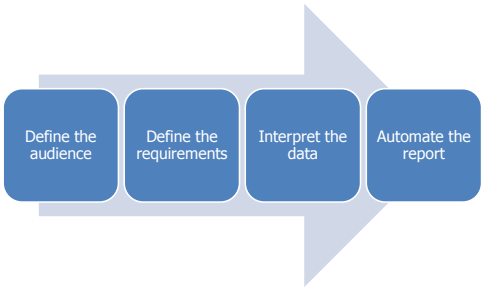
- ❑Report 1: Host Inventory
- ❑Report 2: VM Inventory
- ❑Report 3: Storage Utilization

Creating Great Reports

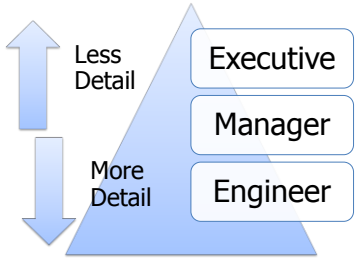
A Special Note About ESXi

- With vCenter, ESXi has good feature parity with ESX
- Without vCenter:
  - Unlicensed (during trial): full functionality
  - Using free license: PowerCLI and other vSphere API consumers are restricted to READ-ONLY access
- Most reporting scenarios work perfectly fine with read-only access!

How to Create Great Reports - A Brief Overview

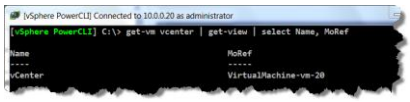


Who Is Your Audience?



What Are Your Requirements?

- Choose an output format
- What fields should you include?
  - VM name: useful
  - VM MoRef: not so useful



Requirements - Output Formats

Format	Cmdlets / Techniques	
CSV	ConvertTo-CSV, Export-CSV	✓
Excel spreadsheet	COM automation, OpenXML	×
Plain text	Format-List, Format-Table	✓
HTML	ConvertTo-Html	✓
Word document	COM automation, OpenXML	×
PDF	Acrobat .NET libraries	×

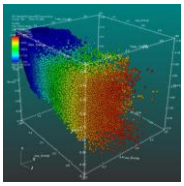
Requirements - Report Destinations

- You have the report -- now what?

Action	Cmdlets / Technique
Email	Send-MailMessage
File Server	Copy-Item
Web Server	Copy-Item (SMB), FTP, SSH

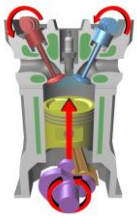
Interpret the Data

- Refrain from repeating data without adding value
- Add value by interpreting, filtering, or analyzing the data
- Combine data from multiple sources
- Enable the reader to draw conclusions from the data
- Easy wins:
  - Add a percent column
  - Indicate threshold violations
- Bonus points:
  - Four-dimensional scatter plot ☺



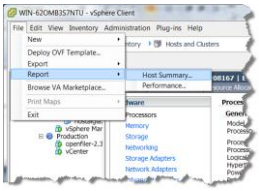
Automate! If You Repeat It, Script It!

- Data collection
- Data interpretation
- Report production
- Send to destination



When Is PowerCLI Useful for Reporting?

- Source data: VMware vSphere
- Need to customize or automate an existing report
- Custom needs not handled by vSphere Client
  - vMotion capability
  - VM detail
  - Storage utilization
  - vSwitch security report
  - Snapshot age



Using PowerCLI to Create Reports

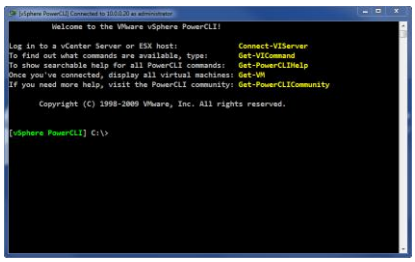
Gather Source Data		
Get-Vm	Get-Vmhost	Get-Snapshot
Interpret the Data		
Measure-Object	Various Operators: + - / *	
Filter, Refine, Format		
Format-List	Format-Table	Select-Object
Export Data		
Export-Csv	ConvertTo-Html	COM Automation
Send to Destination		
Copy-Item	Send-MailMessage	

Actual Scripts!

Report 1: Host Inventory

- Audience: System Engineers
- Requirements
  - Host name
  - Version
  - Number of VMs
  - CPU utilization: max, current, average
  - Memory utilization: max, current, average
  - CPU model
  - Uptime

Report 1: Demo

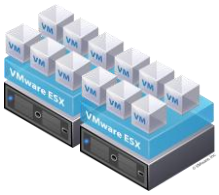


Script: Get-VMHostReport.ps1

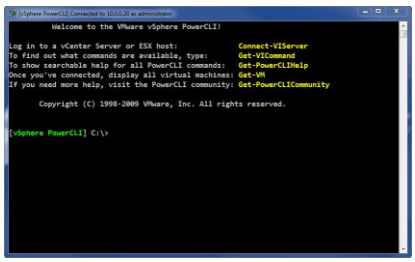
```
param (
    [parameter( Mandatory = $true, ValueFromPipeline = $true )]
    [VMware.VimAutomation.Client20.VMHostImpl]$VMHost
)
process {
    if ( $vmhost.State -ne 'Connected' ) { continue }
    $hv = $vmhost | Get-View
    $HostCpuStat = Get-Stat -Entity $vmhost -stat cpu.usage.average
    $HostMemStat = Get-Stat -Entity $vmhost -stat mem.usage.average
    $span = ( Get-Date ) - $hv.Runtime.BootTime
    $HostProp = @{
        Name            = $vmhost.Name
        Version         = $vmhost.Version
        NumVM           = $hv.Vm.Length
        CpuTotalMhz     = $vmhost.CpuTotalMhz
        CpuUsageMhz     = $vmhost.CpuUsageMhz
        CpuAvgPercent   = $HostCpuStat.Value
        CpuModel        = $vmhost.ProcessorType
        MemTotalMB      = $vmhost.MemoryTotalMB
        MemUsageMB      = $vmhost.MemoryUsageMB
        MemAvgPercent   = $HostMemStat.Value
        HostUptime      = $span.ToString()
    }
    New-Object PSObject -Property $HostProp
}
```


Report 2: VM Inventory

- Audience: Engineer
- Requirements
  - VM Name
  - Total memory allocated
  - Total disk space allocated
  - Number of CPUs
  - Snapshot Information
  - Group by OS
  - Send via email



Report 2: Demo



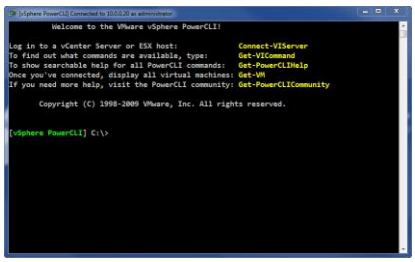
Script: Get-VMReport.ps1

```
<# .SYNOPSIS This script pr...
param (
    [parameter( Mandatory = $true, ValueFromPipeline = $true )]
    [alias("InputObject", "VirtualMachine")]
    [VMware.VimAutomation.Client20.VirtualMachineImpl[]]$VM # Note the extra []'s
)
process {
    $VM | ForEach-Object {
        $vmview = $_ | Get-View
        $snapshot = $_ | Get-Snapshot
        $VMProp = @{
            Name            = $_.Name
            MemoryMB        = $_.MemoryMB
            DiskKB          = ( $_.HardDisks | Measure-Object -Sum CapacityKB ).Sum
            NumCPU          = $_.NumCpu
            NumSnapshot     = if ( $snapshot ) { ( $snapshot | Measure-Object ).Count } else { 0 }
            SnapshotSizeMB  = if ( $snapshot ) { ( $snapshot | Measure-Object -Sum SizeMB ).Sum } else { 0 }
            OSName          = $vmview.Config.GuestFullName
        }
        New-Object PSObject -Property $VMProp
    }
}
```


Report 3: Storage Utilization

- Requirements
  - Average VMDK size
  - Wasted space due to snapshots
  - Wasted space due to abandoned VMDK files
  - Estimate total VM capacity based on current patterns

Report : Demo



[explain script slide]

What We Covered

Creating Great Reports

- ❑How to Create Great Reports - A Brief Overview
  - Who Is Your Audience?
  - What Are Your Requirements?
    - Output Formats
    - Report Destinations
  - Interpret the Data
  - Automate! If You Repeat It, Script It!
- ❑When Is PowerCLI Useful for Reporting?
- ❑Using PowerCLI to Create Reports

What We Covered

Actual Scripts!

- ❑Report 1: Host Inventory
- ❑Report 2: VM Inventory
- ❑Report 3: Storage Utilization

Additional Resources

1. PowerGUI: http://powergui.org/
2. VI SDK Documentation: http://www.vmware.com/support/developer/vc-sdk/visdk400pubs/ReferenceGuide/index.html
3. Powershell data visualization: http://dougfinke.com/blog/?p=468
4. PowerGadgets: http://www.softwarefx.com/sfxSqlProducts/powergadgets/
5. PoshBoard: http://www.poshboard.com/
6. Making graphs with PowerShell & PowerBoots: http://chadwickmiller.spaces.live.com/Blog/cns!EA42395138308430!328.entry
7. VMware datastore graph: http://poshcode.org/588