



VMware vSphere® 5.0 Evaluation Guide

Volume One

TECHNICAL WHITE PAPER
V 1.1/UPDATED AUGUST 11 2011

Table of Contents

About This Guide	5
Intended Audience	5
System Requirements	5
Hardware Requirements	5
Servers	5
Storage	5
Networking	5
Software and Licensing Requirements	6
VMware vSphere	6
Guest Operating Systems	6
Evaluation Guide Environment Setup	7
Server Configuration	8
Logical Network Setup	8
Storage Setup	10
Virtual Machine Setup	10
<i>VMware vSphere 5.0 Evaluation Guide, Volume One</i> – Worksheet	11
vSphere Evaluation Tasks	12
High Availability	12
Introduction	12
Prerequisites	12
Enabling HA	12
Connect to Virtual Server	13
Go to Cluster Summary	14
Edit Cluster Settings	15
Wait for Task to Complete	16
Verifying VMware HA Enablement	16
HA Status Screen	16
Virtual Machine Protection State	20
Host Protection State	21
VMware HA Advanced Options	22
Admission Control	24
Virtual Machine Options	25
Virtual Machine Monitoring	27
Storage Heartbeats	29
Validating VMware HA Operation	29
Host Failure	29

Host Isolation	34
Disabling VMware HA	43
Connect to a Virtual Server	43
Go to the Cluster Summary	44
Edit Cluster Settings	45
Wait for Task to Complete	46
Getting Familiar with the New Command-Line Interface	46
Introduction	46
The New esxcli Command	47
esxcli Command-Line Syntax	47
Remote esxcli Command Authentication	49
Enabling Access to the ESXi Shell	50
Enabling the ESXi Shell from the DCUI	50
Enabling the ESXi Shell from the vSphere Client	50
Enabling SSH Access to the ESXi Shell	51
Enabling SSH from the DCUI	51
Enabling the SSH from the vSphere Client	52
vSphere Client Notification When the ESXi Shell and SSH Are Enabled	53
Installing the vCLI	53
Installing the vCLI on Windows	53
Installing the vCLI on Linux	53
Installing the vCLI with the vMA	53
Sample esxcli Commands Run Locally from the ESXi Shell	54
Sample esxcli Commands Run Remotely from the vCLI	56
Formatting esxcli Output	58
The localcli Command	59
Bringing It All Together	60
vSphere PowerCLI by Example	62
Introduction	62
Prerequisites	62
Install vSphere PowerCLI	62
Getting Started with vSphere PowerCLI	69
Connecting to a vSphere Host or vCenter	69
Using vSphere PowerCLI	71
vSphere PowerCLI Summary	81
Evaluating the ESXi Firewall	81
Introduction	81

Evaluation Overview	81
Prerequisites	81
Stopping SSH Service to Prevent Access	81
Testing Access with SSH Service Stopped	85
Creating Firewall Rules to Block SSH Access.....	86
Testing SSH Firewall Rules	89
Image Builder	91
Introduction.....	91
Image Builder Prerequisites.....	92
Preparation Tasks.....	92
Install vSphere PowerCLI	92
Download the ESXi Offline Bundle	92
Extract the ESXi Offline Bundle	92
Start an Image Builder vSphere PowerCLI Session.....	92
Import the ESXi Offline Bundle	94
Display Software Depots	94
Display VIBs.....	95
Display Image Profiles.....	95
Create a New Image Profile.....	96
Create a New Image Profile by Manually Selecting Individual VIBs	96
Create a New Image Profile by Cloning an Existing Image Profile.....	98
Removing VIBs from an Image Profile	98
Compare Image Profiles.....	99
Export Image Profile.....	99
Export As an Offline Bundle	99
Export As a Bootable ISO Image.....	100
Product Documentation.....	100
Using Storage Performance Statistics	101
Introduction.....	101
Monitoring Performance Statistics of a Datastore.....	103
Help and Support During the Evaluation.....	105
VMware Contact Information	105
Providing Feedback	105

About This Guide

The purpose of the *VMware vSphere 5.0 Evaluation Guide, Volume One*, is to support a self-guided, hands-on evaluation of VMware vSphere® 5.0 (“vSphere”) features usable by all VMware vSphere customers. The companion guide, the *VMware vSphere 5.0 Evaluation Guide, Volume Two*, is intended to highlight vSphere 5.0 features primarily targeted at larger, more complex deployment environments.

Intended Audience

This guide is intended to cover evaluation cases that are suitable for IT professionals who fulfill the following requirements:

- They understand the basics of server virtualization and want to evaluate the features in vSphere in a small-scale deployment.
- They have an existing VMware virtualization environment and want to evaluate features in vSphere that enable greater consolidation while maintaining service levels.

System Requirements

To ensure the best experience when using this guide, the user will need to configure hardware and software as detailed in the following section.

Hardware Requirements

This guide makes the following assumptions about your existing physical infrastructure:

Servers

You must have at least three dedicated servers capable of running VMware ESXi™ 5.0 to provide resources for this evaluation.¹

Storage

You must have shared storage with enough space available to allow the creation of three 100GB dedicated datastores. Shared storage can be SAN or NAS. This document assumes SAN-based storage.

Networking

You must have at least three virtual networks configured to separate virtual machine, vMotion, and vSphere management. These networks can be set up on a single virtual switch with multiple port groups, or across multiple virtual switches. For the purpose of this evaluation guide, the configuration includes a single vSphere standard switch with three port groups.

1. These servers must be on the *VMware vSphere 5.0 Hardware Compatibility List (HCL)*.

For more detailed requirements, see the following table.

HARDWARE	MINIMUM	WHAT'S USED IN THIS GUIDE
ESXi	Three ESXi/ESX servers CPU – Two processors of 2GHz Memory – 6GB Network – 2x 1GB network adaptor	Three ESXi servers (Cisco UCS 1.3.1) CPU – Two quad-core “Nehalem” processors of 2.6GHz Memory – 48GB Network – 4x 10GB network adaptor
Storage	One datastore (100GB)	Three datastores (Fibre Channel – 100GB each)
Network	One VLAN for carrying virtual machine traffic; one VLAN for carrying management traffic	Separate VLANs for ESXi management, vMotion, and virtual machine traffic

Software and Licensing Requirements

This guide makes the following assumptions about your existing software infrastructure:

VMware vSphere

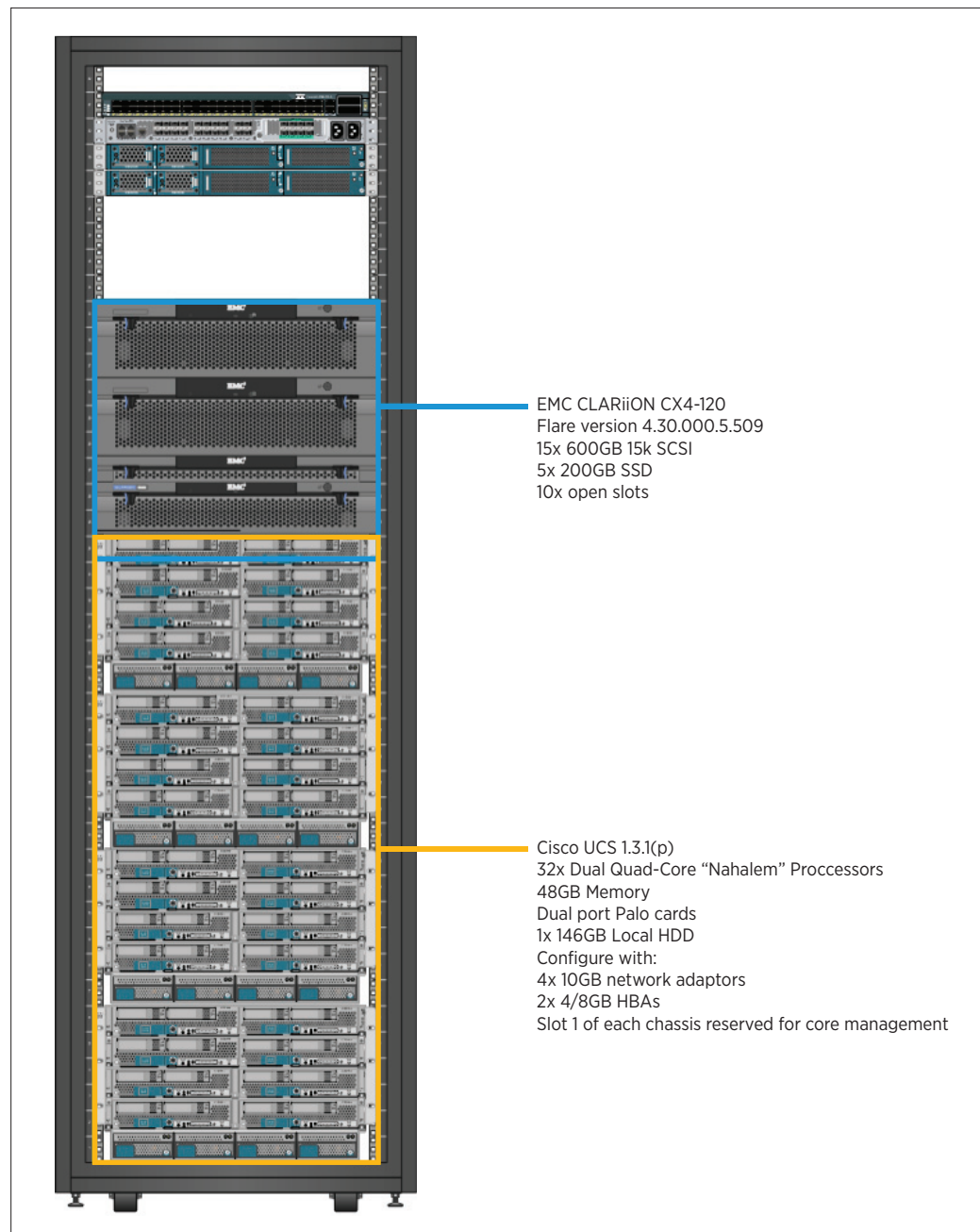
This volume of the *VMware vSphere 5.0 Evaluation Guide* requires vSphere 5.0 and licensing for Essentials Plus. If the user intends to also complete the exercises in Volume Two of the *VMware vSphere 5.0 Evaluation Guide*, a license for Enterprise Plus will be required. The vSphere 5.0 evaluation license available from the VMware evaluation portal provides Enterprise Plus functionality for 60 days and is the best choice for performing the vSphere 5.0 evaluations.

Guest Operating Systems

This volume of the *VMware vSphere 5.0 Evaluation Guide* does not place any specific requirements on guest operating systems, other than ensuring that you can deploy running virtual machines. The user is free to deploy any VMware-supported operating system (OS) in the virtual machines. The *VMware vSphere 5.0 Evaluation Guide, Volume Two*, will require five or six virtual machines running Windows 2003 or Windows 2008.

Evaluation Guide Environment Setup

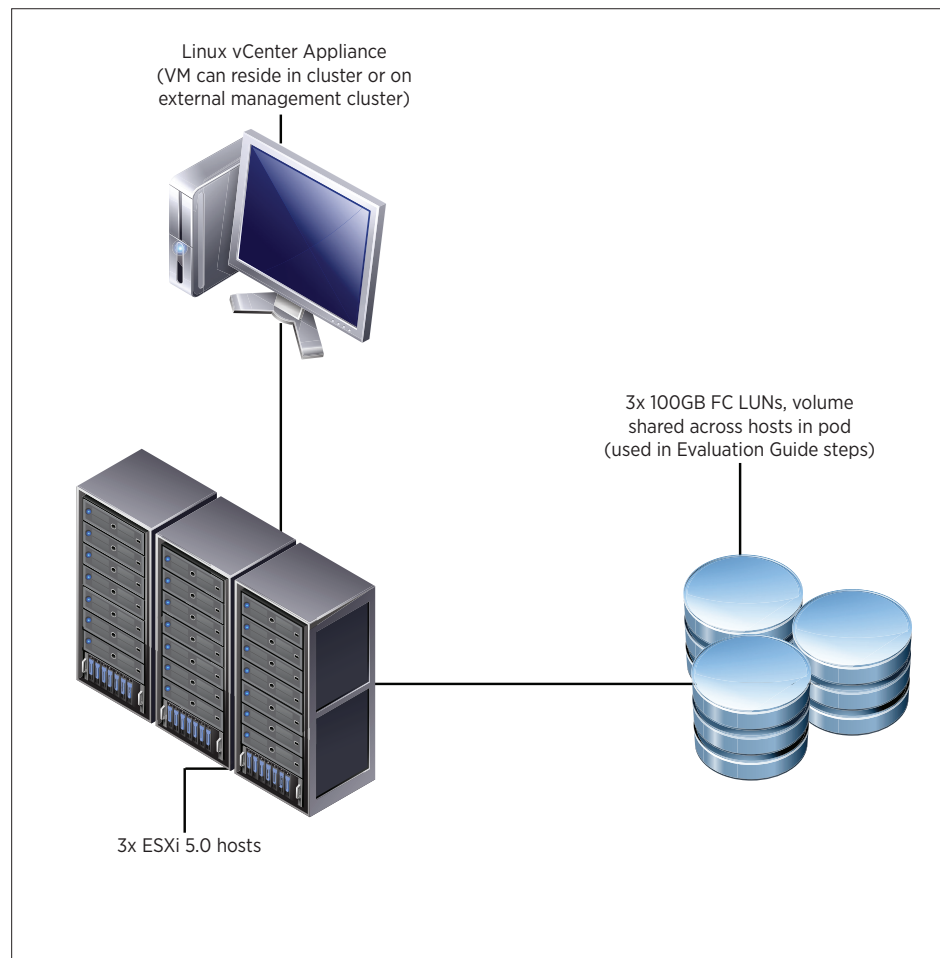
The VMware Technical Marketing lab was built using a combination of Cisco UCS server hardware and EMC CLARiiON CX-4 Fibre Channel (FC) storage. The environment consisted of eight identical four-node “pods,” with most pods configured as a three-node ESXi cluster and a fourth node for management. In many cases, additional resources have been configured in the Technical Marketing test-bed configuration to support other evaluation projects, and are present in the diagrams. The user can configure only what is called for in the following section and can safely ignore additional resources in screen shots and topology diagrams. The following picture shows the Technical Marketing test rack.



Server Configuration

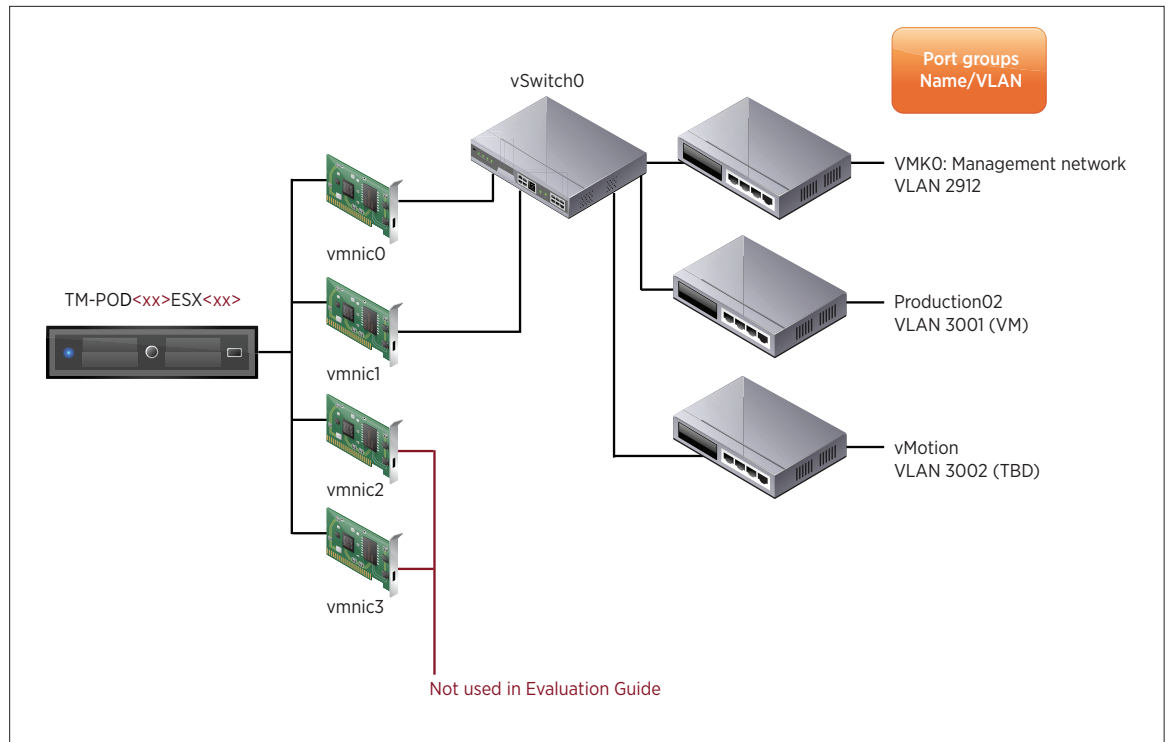
The *VMware vSphere 5.0 Evaluation Guide* calls for three modern server-class systems with adequate processors and memory to host 6–8 minimally configured virtual machines used for testing. The servers used for this evaluation do not need to be overly powerful, just reliable and on the vSphere 5.0 HCL.

Each server must have at least 2x 1GB or 2x 10GB network adaptor and proper connection to shared storage. The following diagram summarizes the evaluation guide test-bed configuration.

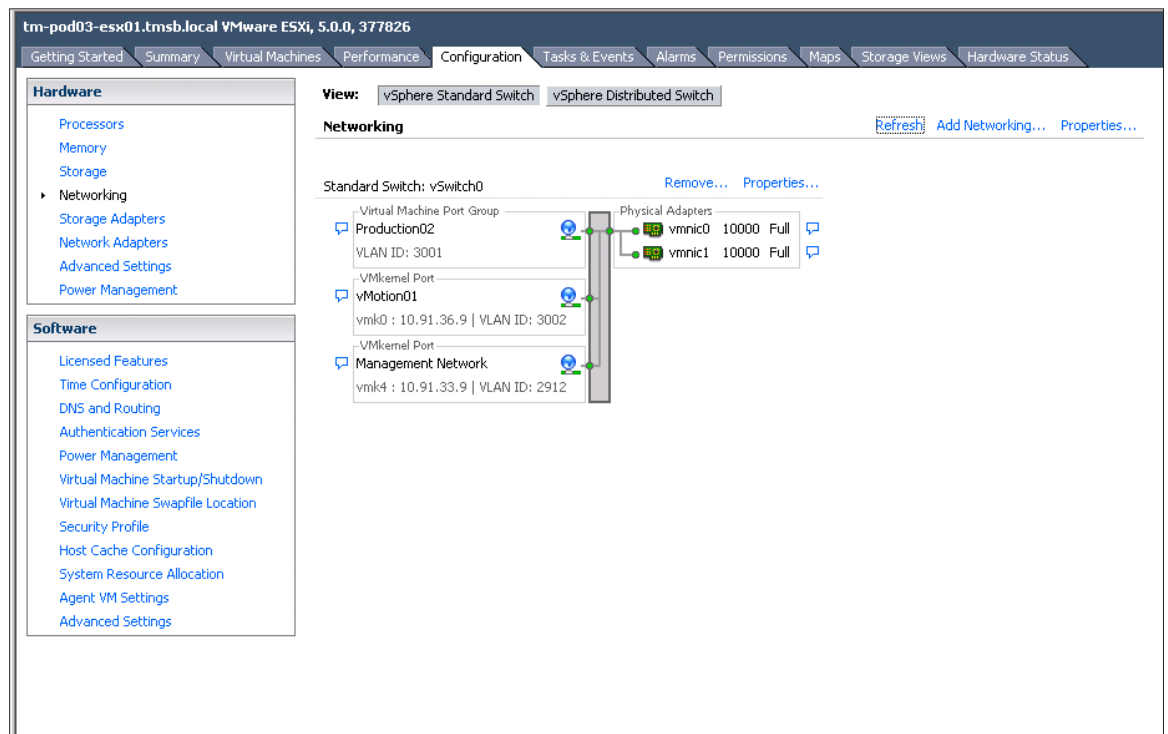


Logical Network Setup

The VMware vSphere 5.0 Evaluation Guide, Volume 1, uses a very simple network configuration consisting of three logical networks. The first is for vSphere management traffic, including vSphere High Availability (VMware HA). The second is for VMware vSphere® vMotion® and the third is for virtual machine traffic. Each logical network is configured as a port group on a standard switch, with a corresponding VLAN configured to provide physical isolation of the network traffic.

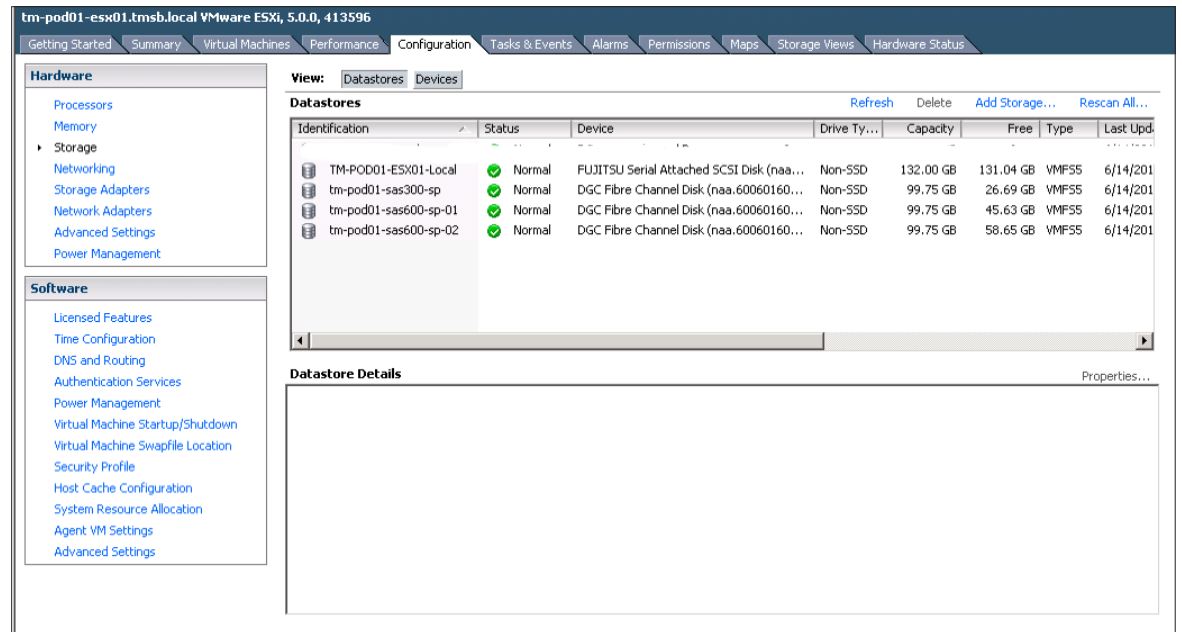


On the vSphere side, the network configuration looks like the following:



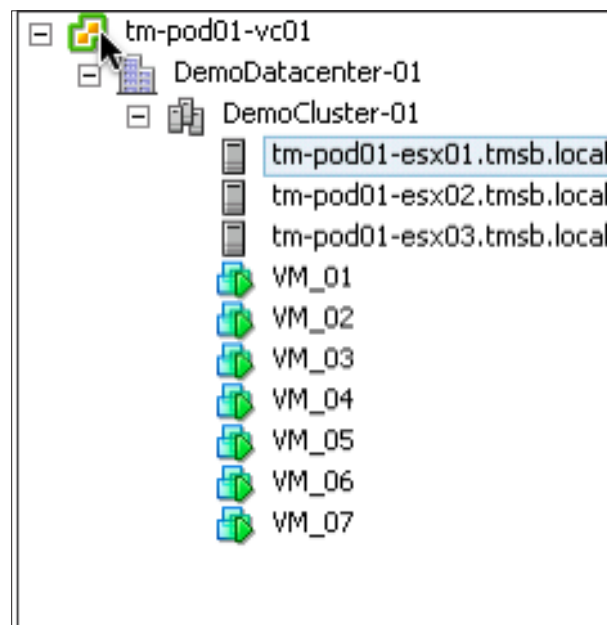
Storage Setup

The *VMware vSphere 5.0 Evaluation Guide, Volume One*, uses a storage configuration consisting of three 100GB FC LUNs presented to each host, enabling creation of three datastores.



Virtual Machine Setup

The *VMware vSphere 5.0 Evaluators Guide, Volume One*, uses a total of six to seven virtual machines for testing. These can be Linux or Windows virtual machines. It is up to the user to configure virtual machines that can be brought up to a running state for testing. The following diagram shows VM_01 through VM_07 configured in the Technical Marketing test lab:



VMware vSphere 5.0 Evaluation Guide, Volume One – Worksheet

You can use the following worksheet to organize your evaluation process.

HARDWARE CHECKLIST:	
All hardware has been validated against the <i>VMware vSphere 5.0 Hardware Compatibility List (HCL)</i> .	
Each server has 2x 1GB or 2x 10GB network cards connected to a common switch (this will be configured as a network adaptor team).	
Each server has the required HBA/network adaptor to access shared storage.	

SOFTWARE CHECKLIST:	
VMware vSphere/VMware ESXi installation media is available.	
VMware vCenter™ Server appliance is downloaded.	
VMware vSphere® Client™ is installed.	
ESXi host 1 hostname.	
ESXi host 2 hostname.	
ESXi host 3 hostname.	
Subnet, netmask and default gateway for management network.	
Subnet, netmask and default gateway for virtual machine network.	
Subnet, netmask and default gateway for vMotion network.	

STORAGE CHECKLIST:	
All servers can see at least three common 100GB LUNs (or NFS exports).	
Datastore 1 name.	
Datastore 2 name.	
Datastore 3 name.	

vSphere Evaluation Tasks

High Availability

Introduction

Ensuring the availability of virtual machines within an environment is of paramount concern to administrators. VMware HA alleviates these concerns by providing protection from failures within the following three key layers:

- **The infrastructure layer**
At this layer, VMware HA monitors the health of the virtual machine and will attempt to restart the virtual machine when a failure, such as the loss of a physical host, occurs. This protection is independent of the OS used within the virtual machine.
- **The OS layer**
Through the use of VMware Tools installed within the OS, VMware HA can monitor the OS for proper operation. This protects against such failures as an unresponsive OS.
- **The application layer**
With some customization or with a third-party tool, an administrator can also monitor the application running within the OS for proper operation. In the event of a failure of the application, HA can be triggered to restart the virtual machine hosting the application.

In this section, you will learn how to enable, configure, and test the operation of HA to provide basic high availability services for your virtual machines at the infrastructure layer.

Prerequisites

Before continuing, it is important that the environment be configured properly. Refer to the “System Requirements” section of this document and verify that the environment you are using is configured as documented. Specific areas of interest include the following:

- Ensure that you have a working management network with all hosts in the environment.
- Verify that all of the virtual machines are online.
- Have at least one virtual machine running on each host.
- Validate that you have access to VMware vCenter™ utilizing the vSphere Client.

Enabling HA

Enabling HA is a straightforward process that simply entails editing the properties for the cluster. The following steps will guide you through this process.

Connect to Virtual Server



Figure 1. Connecting to Virtual Server

Using the vSphere Client, connect to your virtual server instance.

Go to Cluster Summary

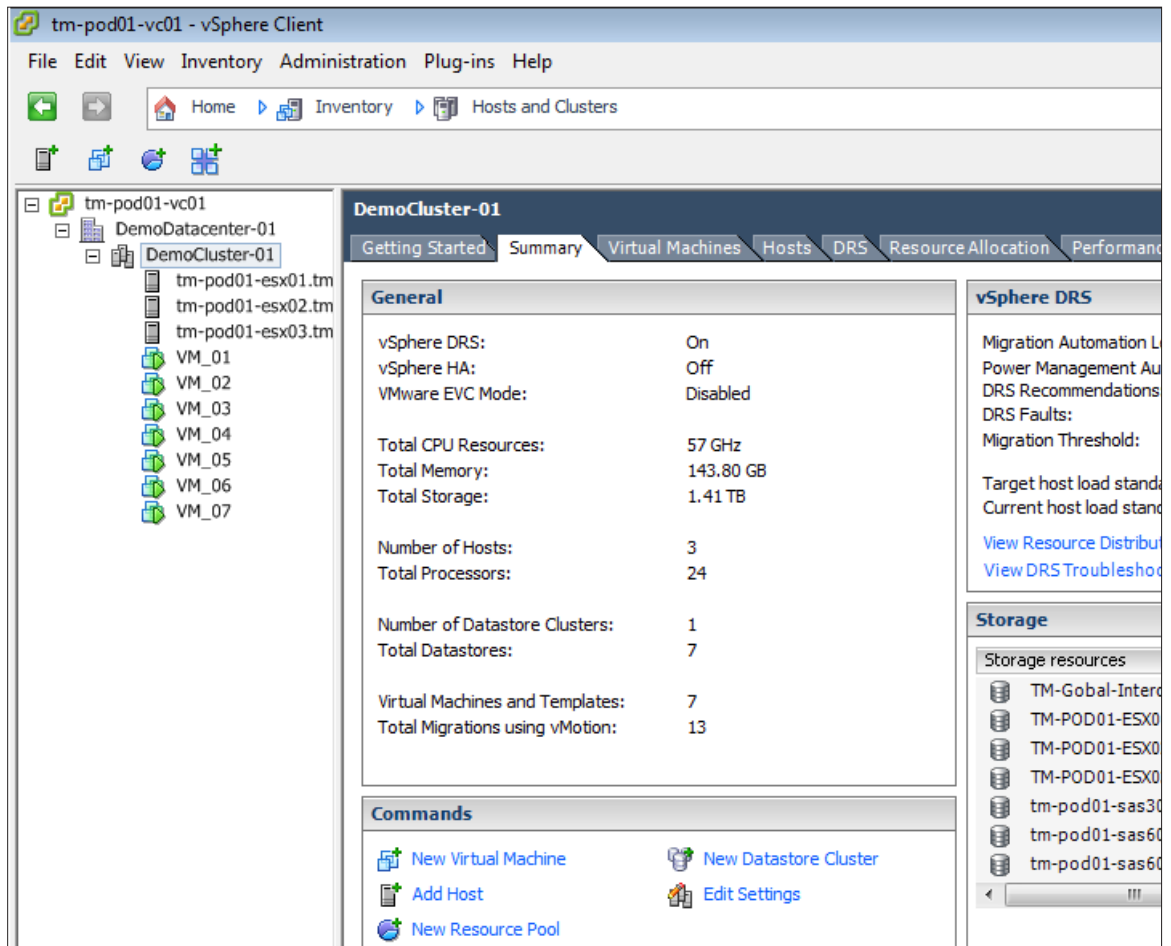


Figure 2. Cluster Summary

Once connected to your virtual server instance, select your cluster by clicking on its name on the left-hand panel. Select the **Summary** tab to bring up the cluster summary screen.

Edit Cluster Settings

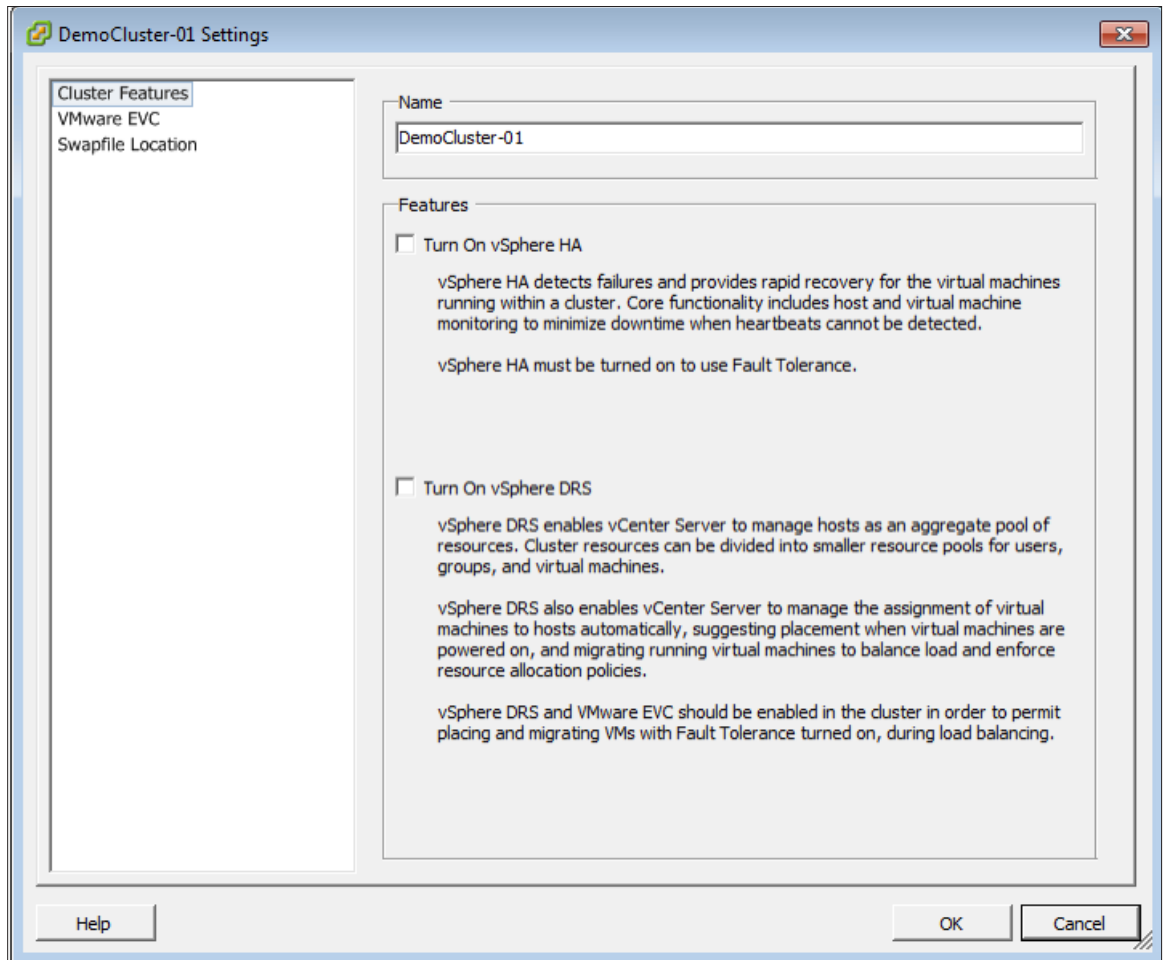


Figure 3. Editing Cluster Settings

In the cluster summary screen, select the **Edit Settings** option. This will bring up a wizard that you can use to modify the settings of the cluster. Click the check box next to **Turn On vSphere HA** and select OK. This will close the wizard and the system will initialize VMware HA.

Recent Tasks		
Name	Target	Status
Configuring vSphere HA	tm-pod01-esx02.tmsb.local	50%
Configuring vSphere HA	tm-pod01-esx03.tmsb.local	In Progress
Configuring vSphere HA	tm-pod01-esx01.tmsb.local	In Progress

Figure 4. Initializing VMware HA

Under the Recent Tasks pane of the vSphere Client, you can observe the progress of the initialization of HA on the systems within the cluster. You'll notice that the configuration tasks occur in parallel among all the hosts within the cluster.

Wait for Task to Complete







Name	Target
 Configuring vSphere HA	 tm-pod01-esx02.tmsb.local
 Configuring vSphere HA	 tm-pod01-esx03.tmsb.local
 Configuring vSphere HA	 tm-pod01-esx01.tmsb.local

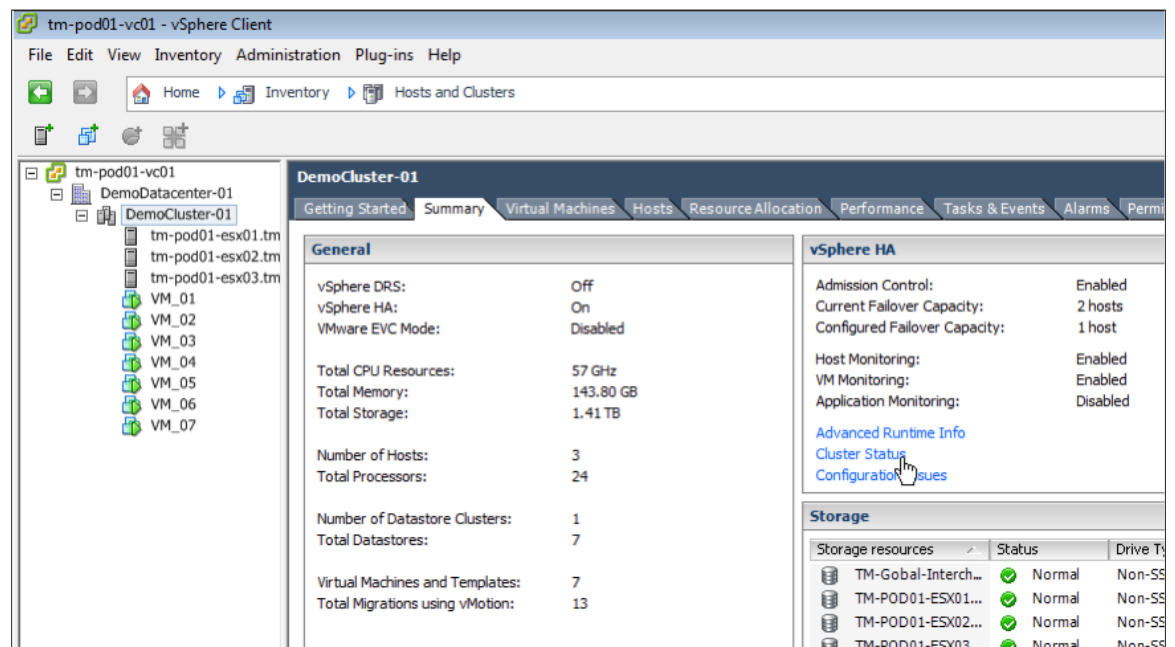
Figure 5. Tasks Showing Completed Status

Wait until all the tasks show a **Completed** status. This should only take a minute. At this point, VMware HA is now providing protection for the virtual machines that are powered on.

Verifying VMware HA Enablement

At this point, VMware HA should be enabled within your cluster. This section will demonstrate several methods you can use to verify that HA is enabled.

HA Status Screen



The screenshot shows the vSphere Client interface for 'tm-pod01-vc01'. The left pane shows the inventory tree with 'DemoCluster-01' selected. The main pane shows the 'Summary' tab for 'DemoCluster-01'. The 'vSphere HA' section is expanded, showing the following settings:

Setting	Value
vSphere DRS	Off
vSphere HA	On
VMware EVC Mode	Disabled
Total CPU Resources	57 GHz
Total Memory	143.80 GB
Total Storage	1.41 TB
Number of Hosts	3
Total Processors	24
Number of Datastore Clusters	1
Total Datastores	7
Virtual Machines and Templates	7
Total Migrations using vMotion	13

The 'vSphere HA' section also shows the following settings:

Setting	Value
Admission Control	Enabled
Current Failover Capacity	2 hosts
Configured Failover Capacity	1 host
Host Monitoring	Enabled
VM Monitoring	Enabled
Application Monitoring	Disabled

Below these settings, there are links for 'Advanced Runtime Info', 'Cluster Status', and 'Configuration Issues'. The 'Cluster Status' link is highlighted with a mouse cursor.

The 'Storage' section shows a table of storage resources:

Storage resources	Status	Drive Type
TM-Gobal-Interch...	Normal	Non-SS
TM-POD01-ESX01...	Normal	Non-SS
TM-POD01-ESX02...	Normal	Non-SS
TM-POD01-ESX03...	Normal	Non-SS

Figure 6. Configuration of HA

After enabling HA, you will notice that a section for HA is now shown under the cluster summary screen. This will show you general information about the configuration of HA. There is also an option for **Cluster Status** here. Click this to bring up the HA Cluster Status screen.

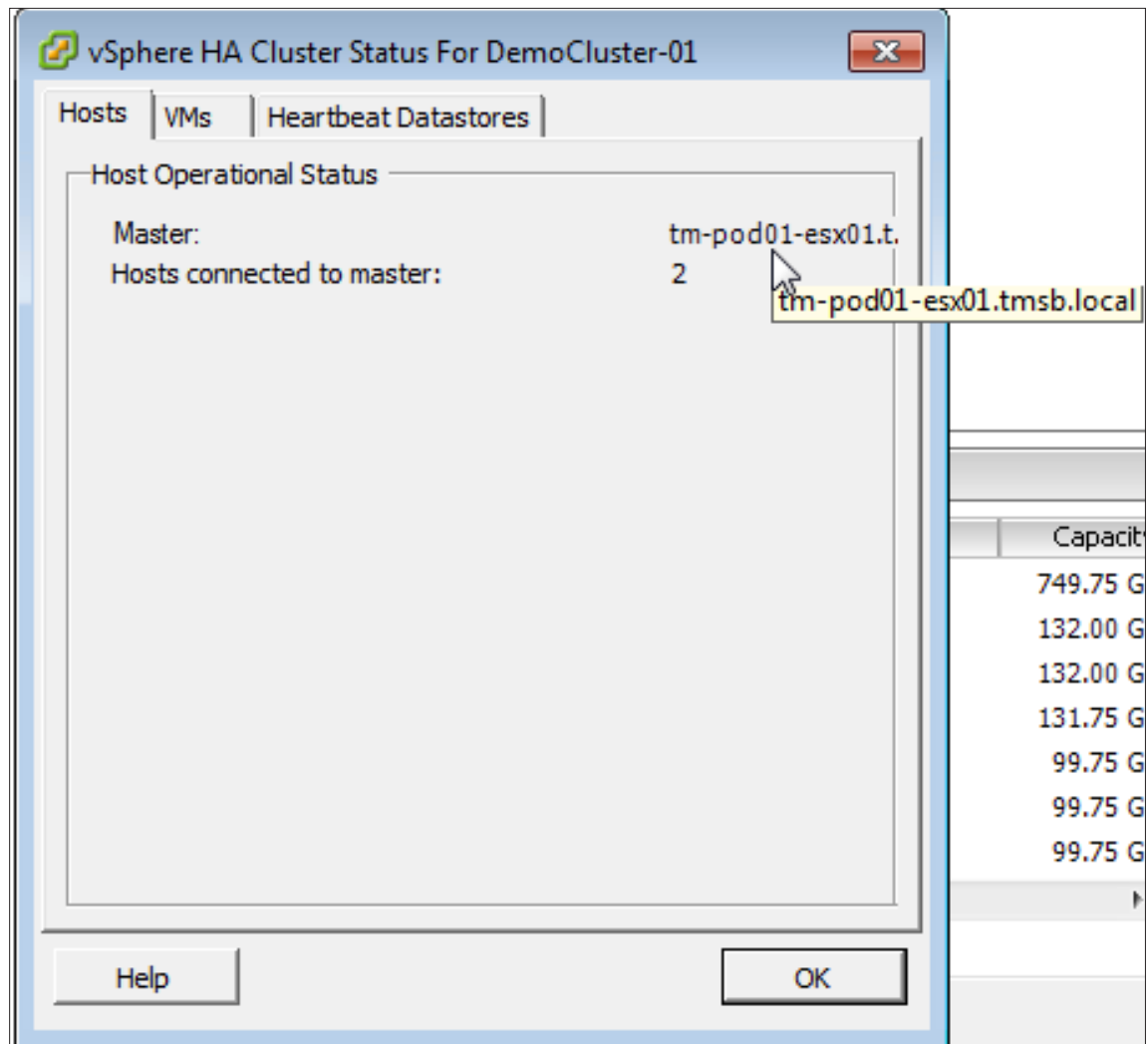


Figure 7. VMware HA Cluster Status Screen

Under this screen, you will notice three tabs. There is one tab each for Hosts, VMs, and Heartbeat Datastores. On the Hosts tab, you will see the system that is acting as the Master node. You will also see the number of hosts that are currently connected to this Master. The number shown should equal the number of hosts that are contained within your cluster, minus one for the Master.

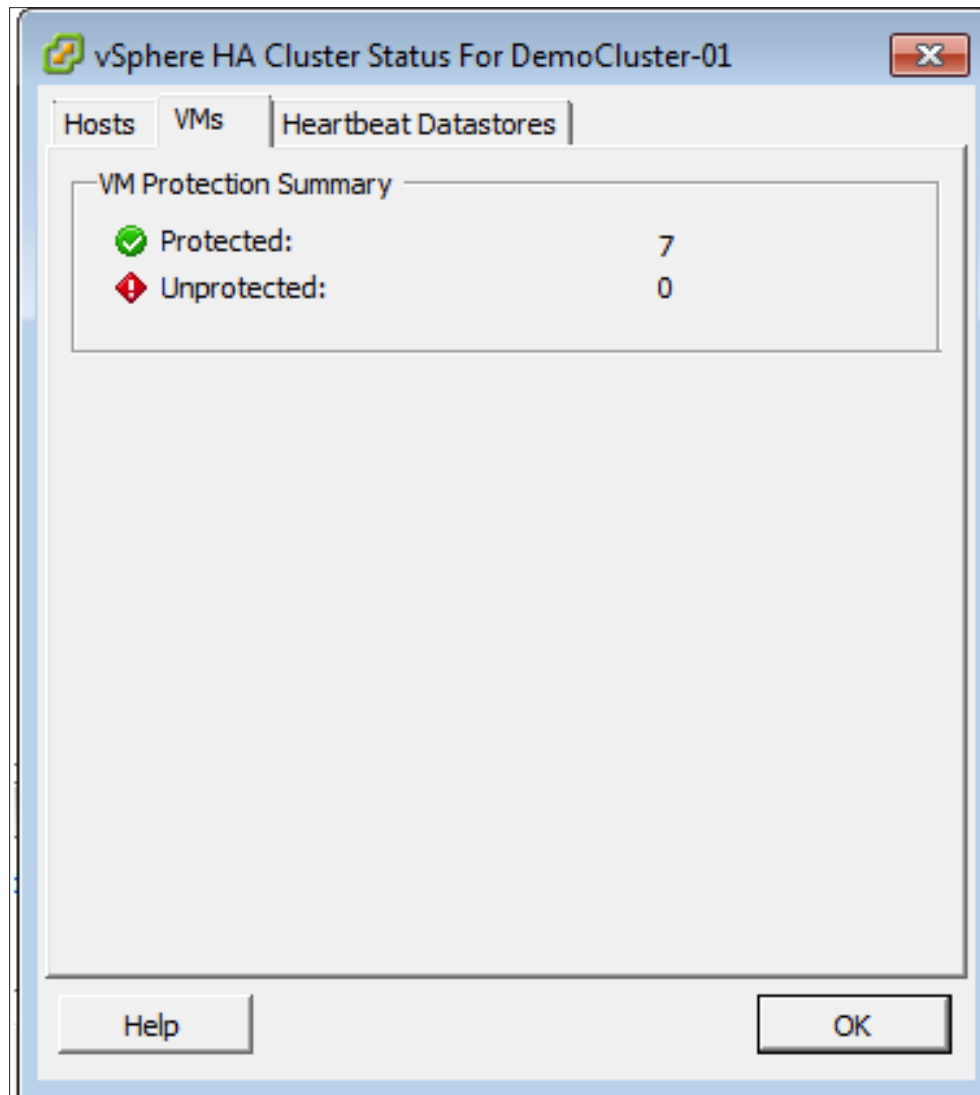


Figure 8. Summary of Virtual Machine Protection States

Under the **VMs** tab, a summary of the virtual machine protection states is displayed. The virtual machines that were powered on when VMware HA was enabled are in the **Protected** state.

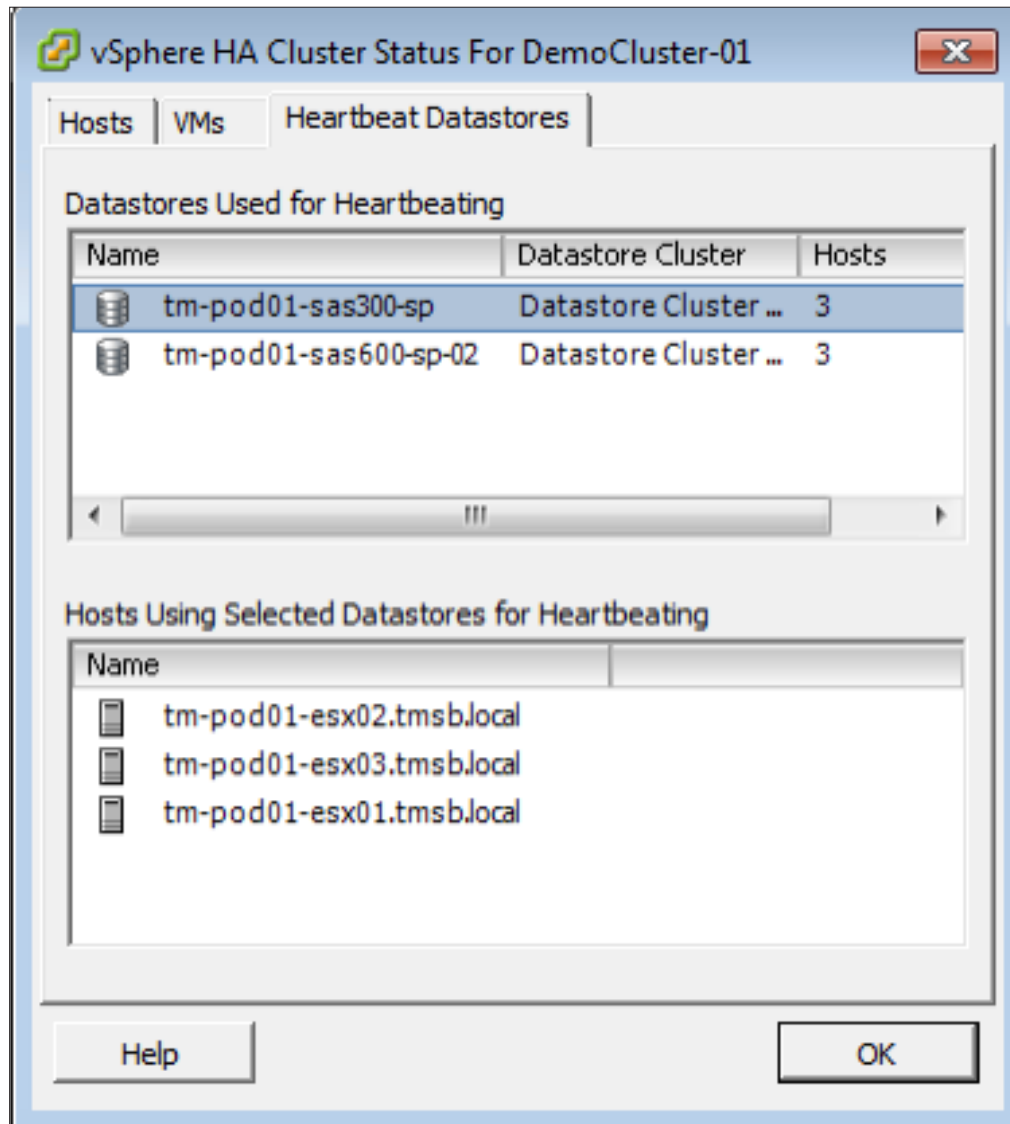


Figure 9. Heartbeat Datastores Information

Clicking the Heartbeat Datastores tab will display information about the datastores that were selected as heartbeat datastores. Heartbeat datastores allow a secondary means of communication between the hosts in case of a loss of the management network. By selecting a particular datastore, you will display a list of all the hosts that are using the selected datastore as a heartbeat datastore.

Click **OK** to exit the cluster status screen.

Virtual Machine Protection State

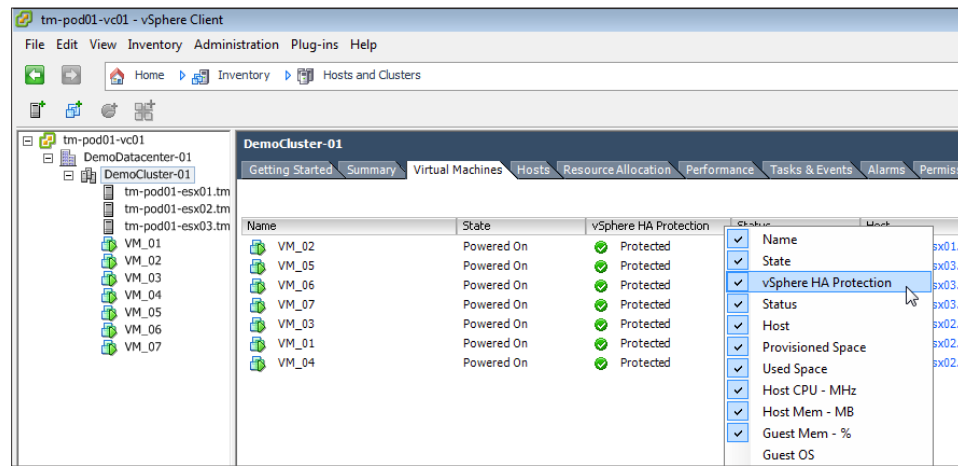


Figure 10. Viewing the Current VMware HA Protection State

Another method you can use to see the protection state of the virtual machines would be to select the **Virtual Machines** tab for a cluster. Right-clicking the title bar enables you to select the **vSphere HA Protection** field. Once the field is selected, you will see a column that displays the current VMware HA protection state for every virtual machine within the cluster.

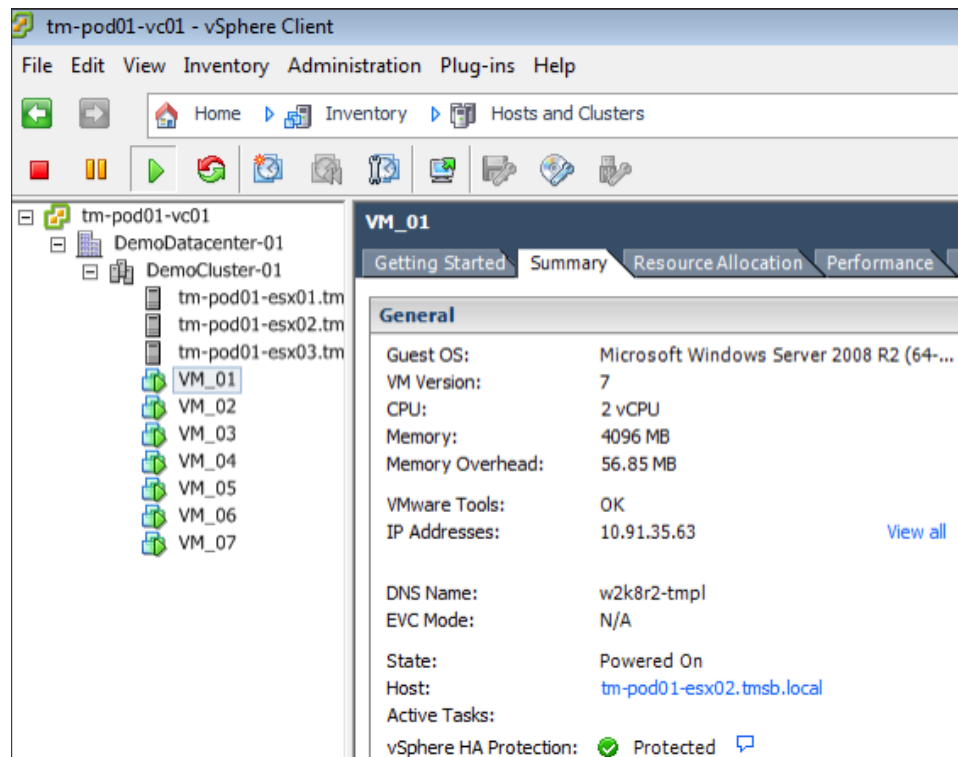


Figure 11. HA Protection State for an Individual Virtual Machine

You can also identify the HA protection state for an individual virtual machine by selecting the virtual machine on the navigation tree and then clicking the Summary tab.

Host Protection State

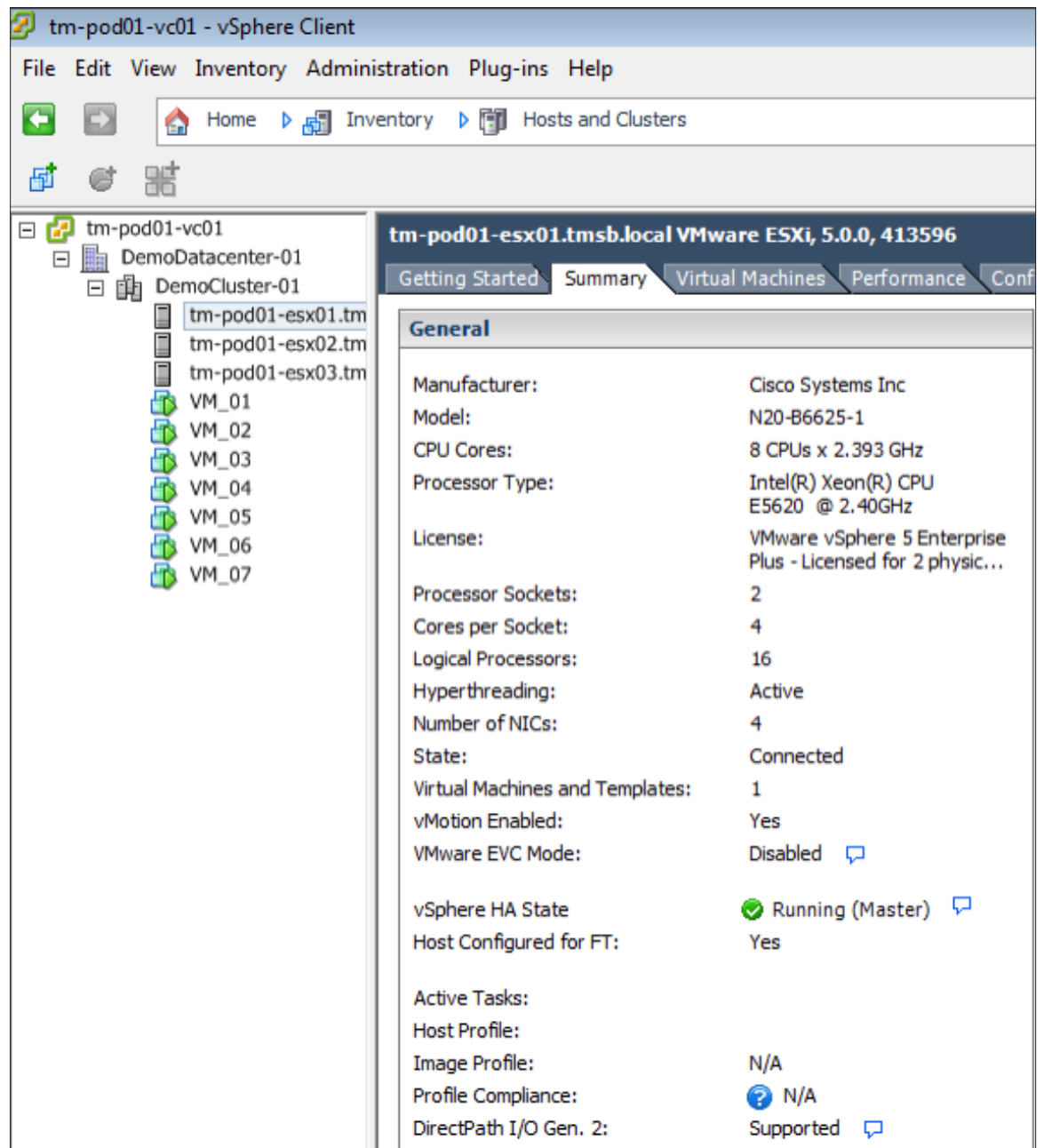


Figure 12. Viewing the VMware HA State for the Host

The VMware HA state can be identified for an individual host by selecting the desired host from the navigation tree and selecting the Summary tab. Here you will see the VMware HA state for the host as well as the role that this node plays within the cluster. In the preceding example, the host is the master node for the cluster.

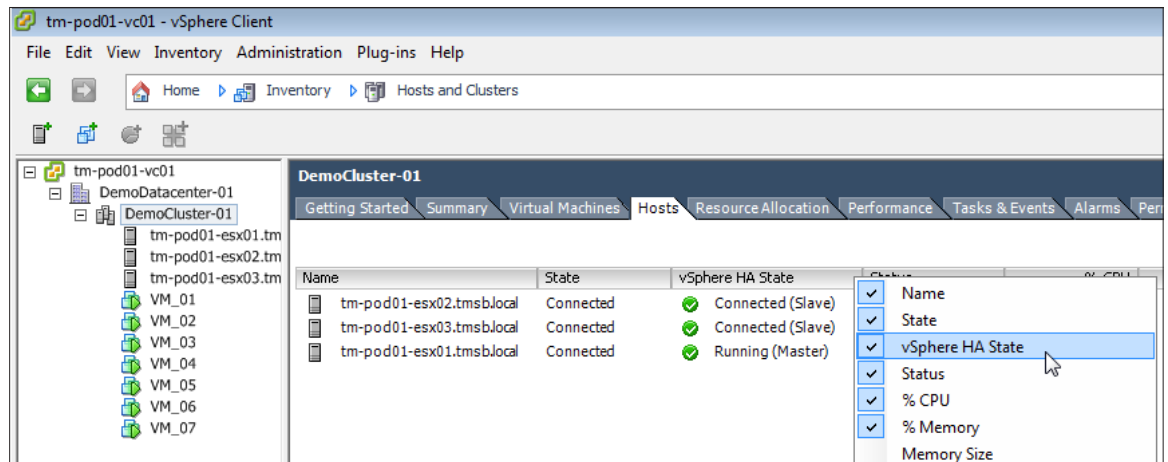


Figure 13. Displaying the HA State for All Hosts Within a Cluster

To display the VMware HA state for all of the hosts within a cluster, select the cluster from the navigation tree and then click the Hosts tab. Right-click the title bar and ensure that the vSphere HA State column is enabled.

VMware HA Advanced Options

VMware HA provides a user with the ability to change various options based on their individual needs. This section provides an overview of the most commonly used options.

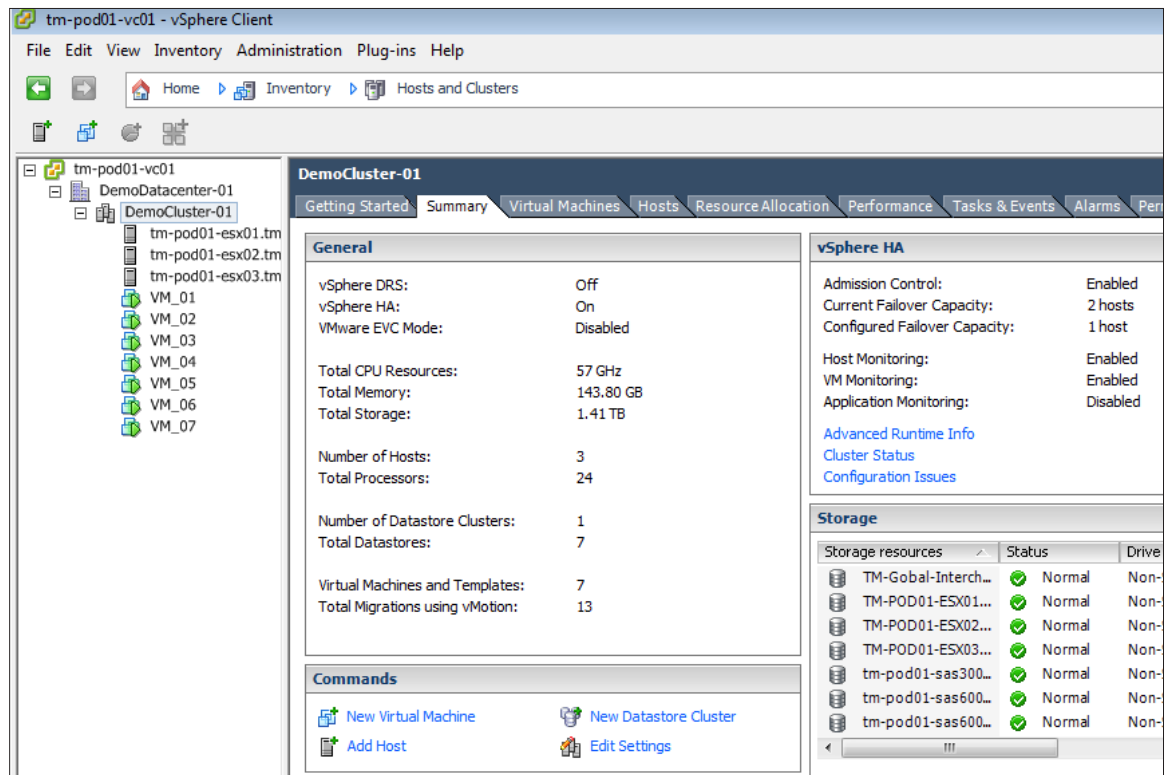


Figure 14. Editing Settings

Click **Edit Settings**.

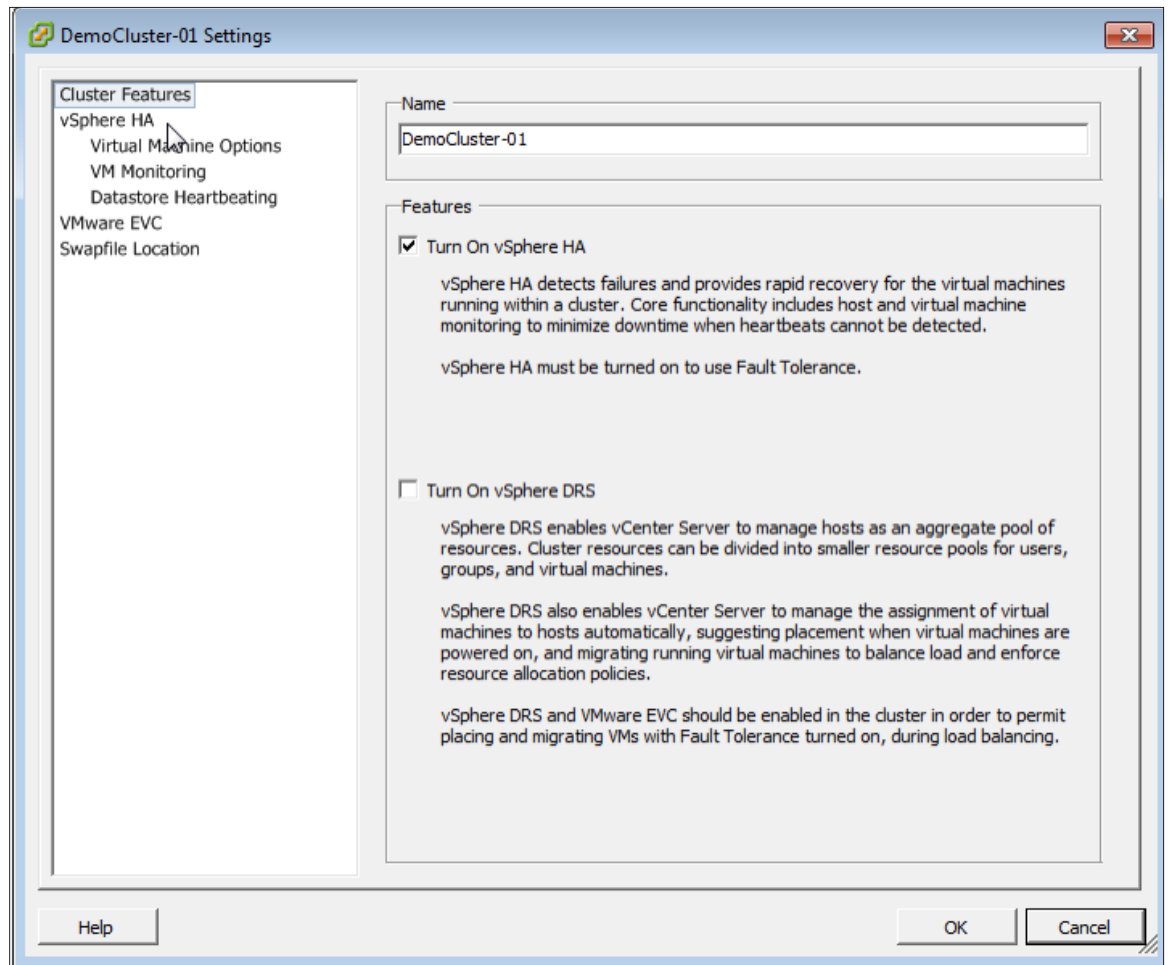


Figure 15. Cluster Settings Wizard

This brings up the wizard that allows you to edit the cluster settings. Once VMware HA is enabled, additional settings are displayed allowing for the configuration of VMware HA.

Admission Control

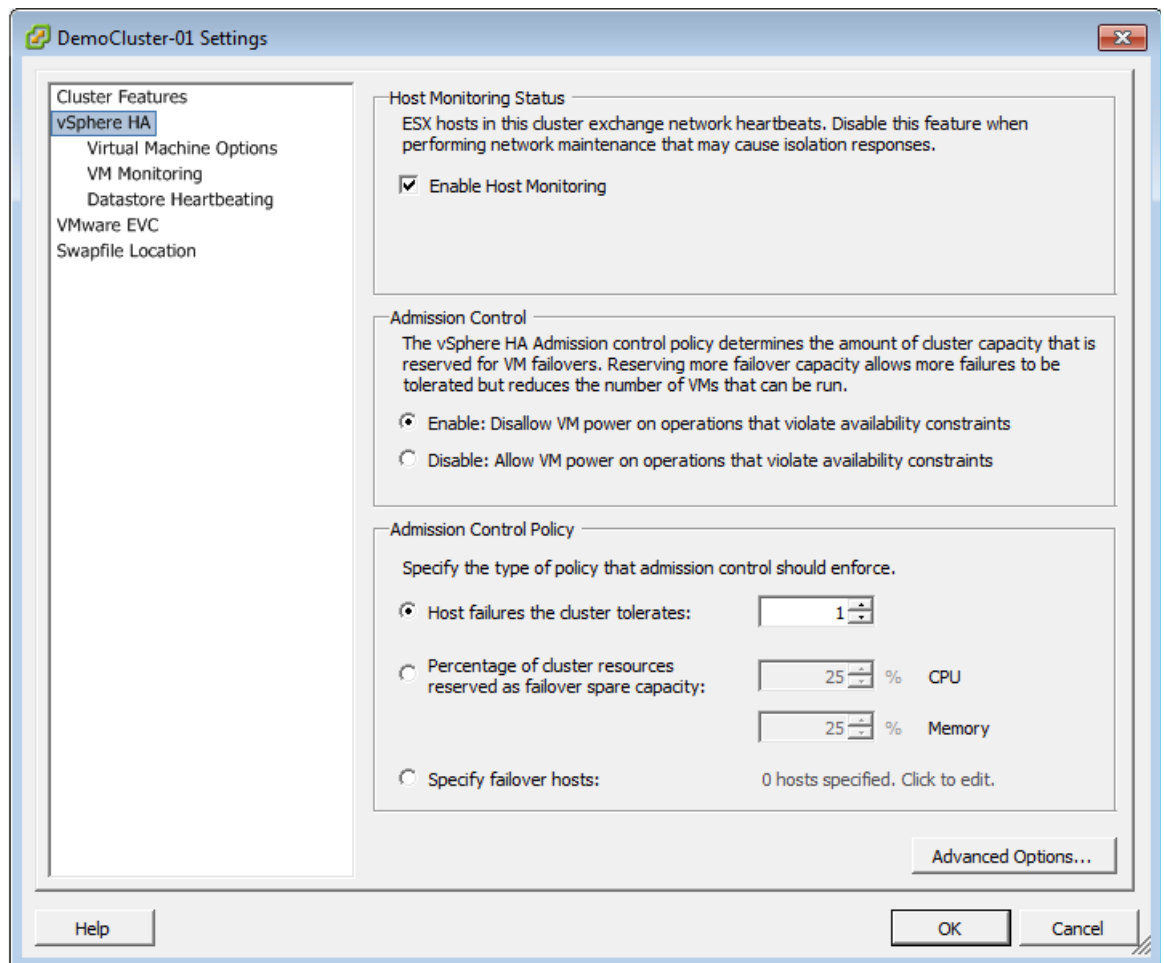


Figure 16. Host Monitoring Status and Admission Control Attributes

In the cluster settings dialog box, select vSphere HA from the navigation tree on the left. This allows you to edit the Host Monitoring Status and Admission Control attributes.

Host monitoring enables VMware HA to take action if a host fails to send heartbeats over the management network. During maintenance operations on the management network, it is possible that the hosts will not be able to send heartbeats. When this occurs, you should unselect this option to prevent VMware HA from believing the hosts are isolated.

Admission control is used to ensure that adequate resources within the cluster are available to facilitate failover if needed. It also serves to ensure that the virtual machine reservations are respected. Three options are available to specify the desired admission control policy. These include the following:

- **Host failures**
This option attempts to reserve enough capacity within the cluster to provide for the failure of any host within the cluster.
- **Percentage**
As with the host failures option, this also attempts to reserve enough capacity within the cluster. However, this option allows you to specify a percentage of CPU and memory that you want reserved.

- Failover hosts

Alternately, you can specify particular hosts within the cluster that will be used as a preferred target host to start any virtual machines that were protected on a failed host. In the event of a failure, vSphere HA will first attempt to restart the protected VMs on these hosts before trying others. Additionally, vSphere HA prevents VMs from being moved to these hosts, or powered on by the user or vSphere Distributed Resource Scheduler (DRS) on these hosts.

Virtual Machine Options

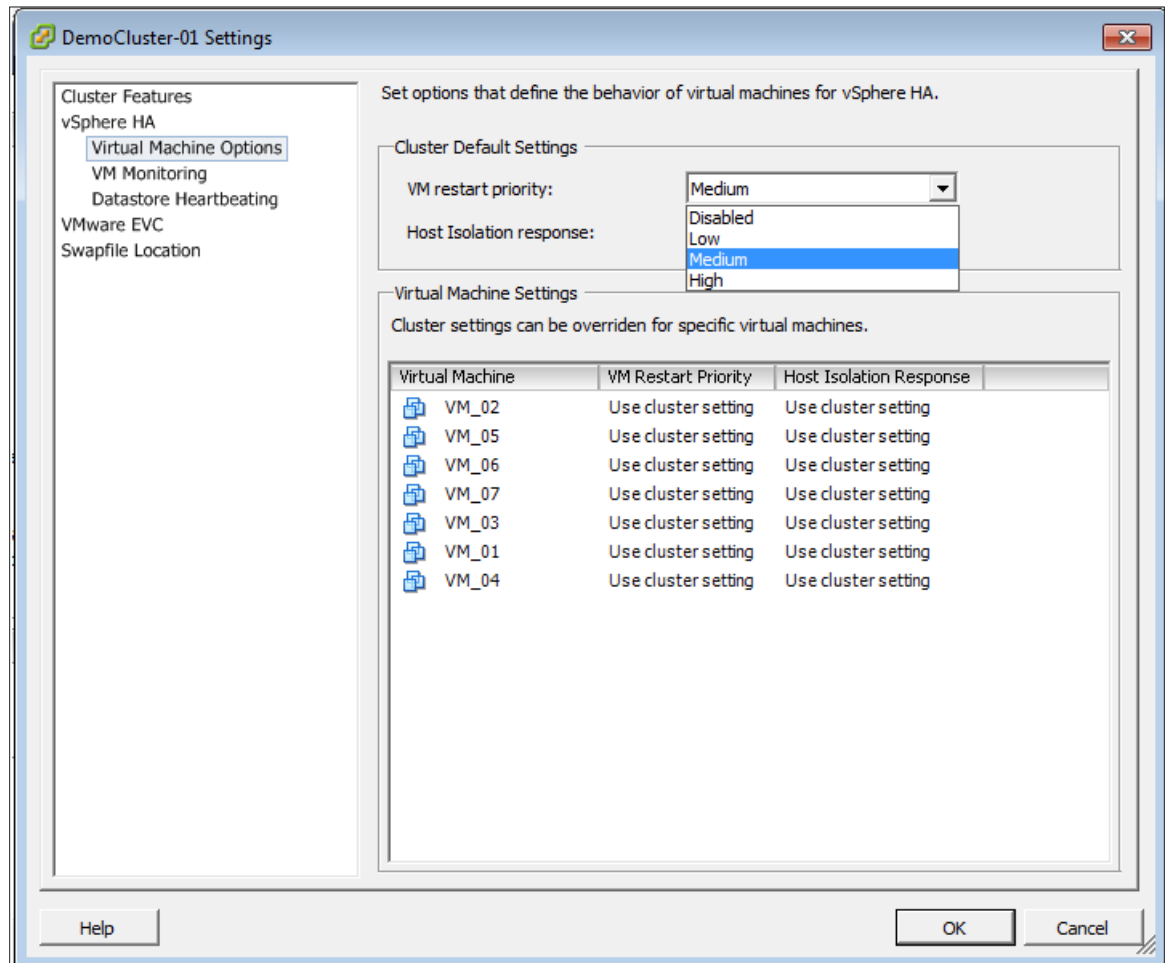


Figure 17. Defining the Behavior of Virtual Machines for VMware HA

Select Virtual Machine Options from the left-hand navigation pane. Here, you can define the behavior of virtual machines for VMware HA. The two settings you can edit are the VM restart priority and the Host Isolation response.

The VM restart priority enables you to specify the order that virtual machines will be started in the event of a failure. In cases where there might not be enough resources available within the cluster to accommodate the restart of a series of virtual machines, this setting allows a level of prioritization, allowing the most important virtual machines to be restarted first. Notice that this can be set on a per-virtual machine basis as well.

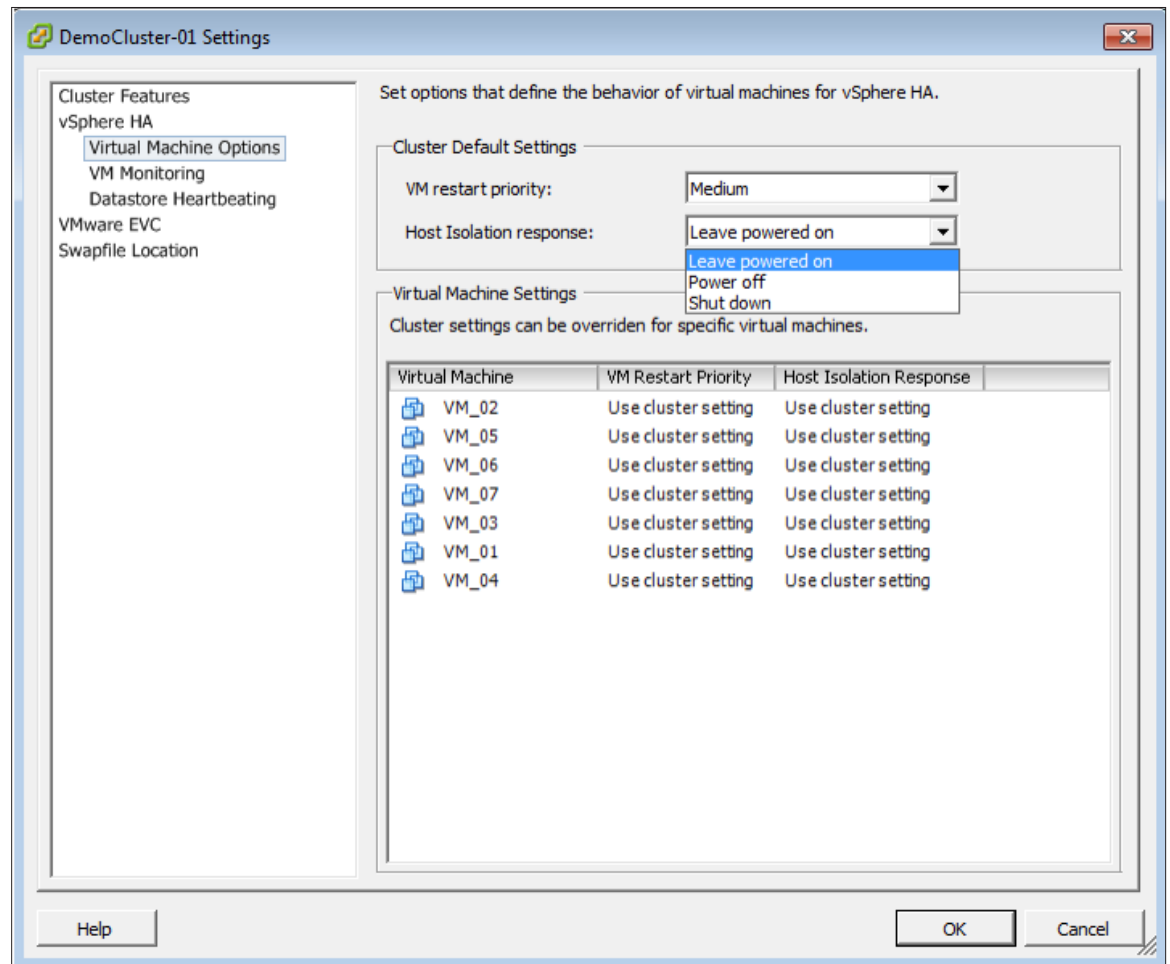


Figure 18. Host Isolation Response

Host Isolation Response specifies the behavior that HA will take in the event that a host is determined to be isolated. Host isolation occurs when a host loses the ability to communicate through the management network to the other hosts within the environment and is unable to ping its configured isolation addresses—this is the default gateway. In this event, the host is still functioning, although it is not able to communicate. The default setting for this is **Leave powered on**.

Virtual Machine Monitoring

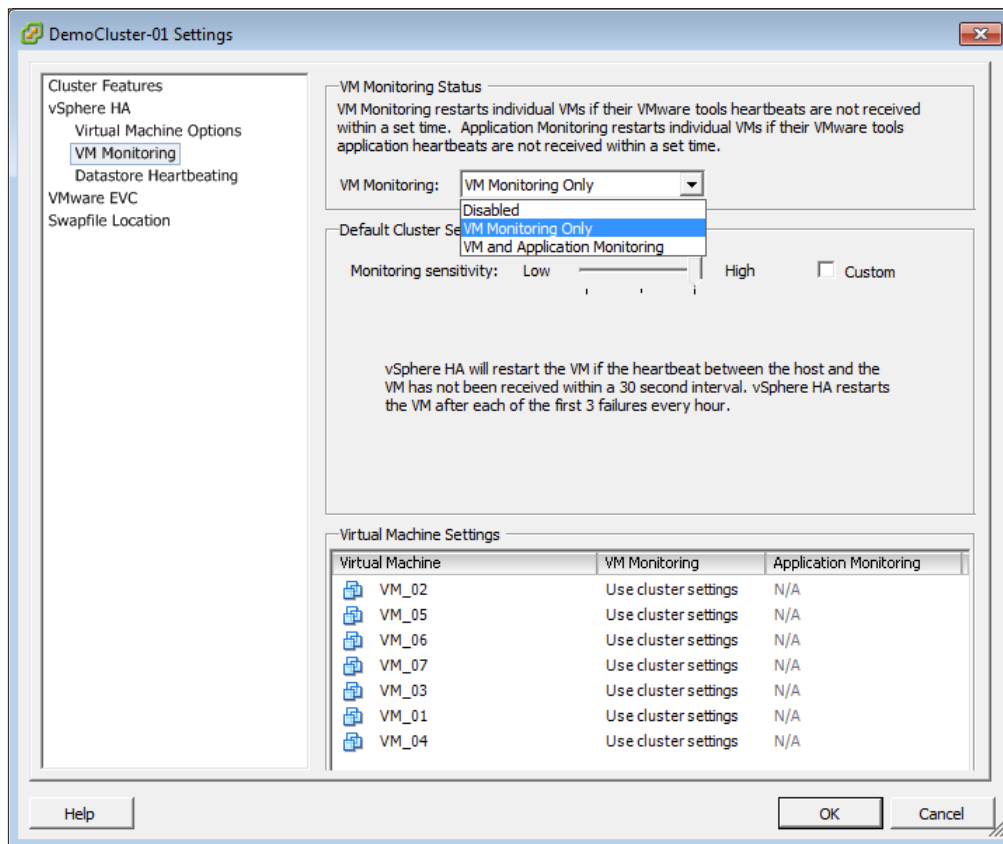


Figure 19. Virtual Machine Monitoring

Selecting VM Monitoring from the left-hand navigation pane enables you to change settings related to the monitoring of the OS or application running within a virtual machine. In order to use this feature, you must have VMware Tools installed within the virtual machine.

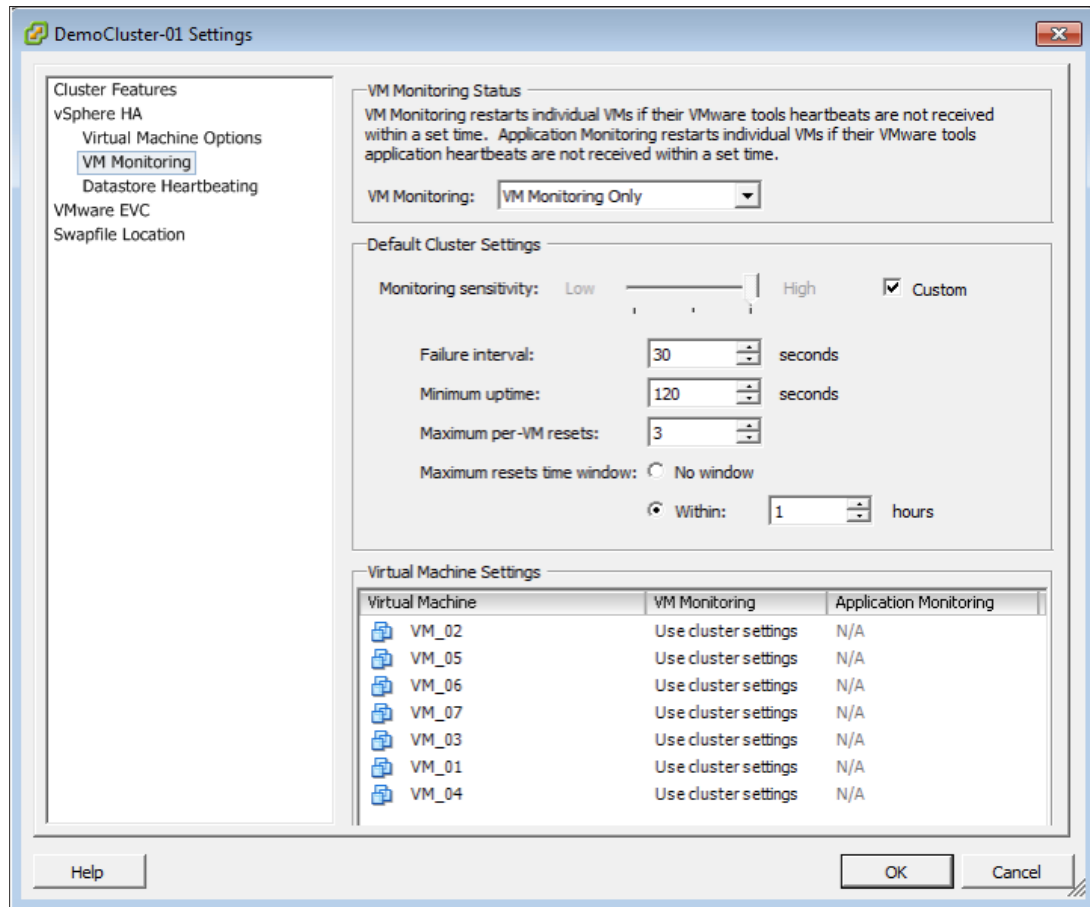


Figure 20. Selecting Custom Option for VM Monitoring

By selecting the Custom option, you can exert a fine level of control over the various parameters involved. You can specify these settings on a per-virtual machine basis.

Storage Heartbeats

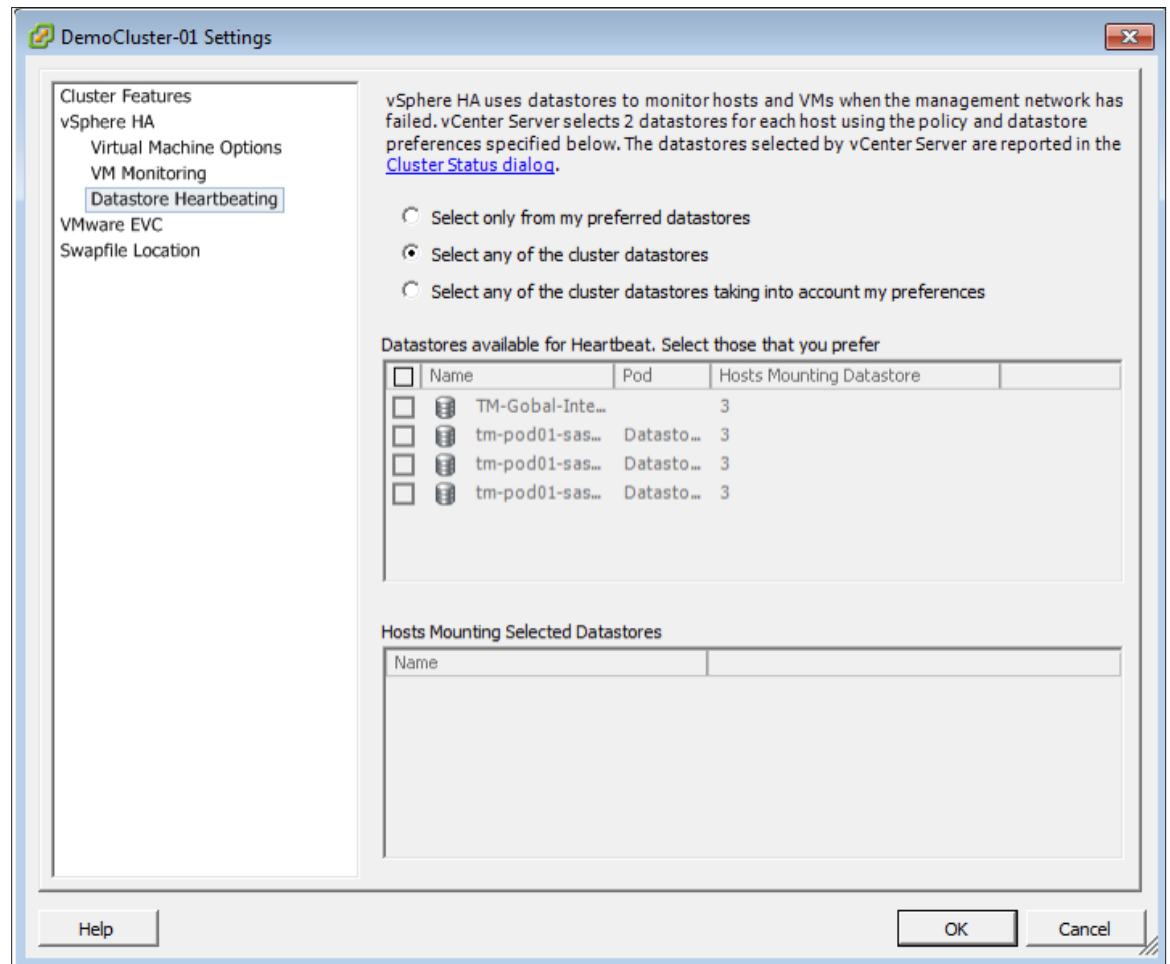


Figure 21. Datastore Heartbeating Window

Storage heartbeats provide a secondary communication path in the event of a failure of the management network. This is advantageous, because it provides another level of redundancy and allows for the determination of failure between a network and a host failure. By default, two datastores will be chosen based on the connectivity they have to other hosts and the type of storage. This attempts to provide protection against array failures and allows for the highest number of hosts to utilize the heartbeat datastore. The datastores utilized can be manually specified if desired.

Validating VMware HA Operation

In order to see VMware HA in action, we need to inject faults into the environment. This section will demonstrate the ways in which to do this for the most common failure cases, so that you can validate the operation of VMware HA and can test ways to recover from a failure.

Host Failure

The most common failure case involves the failure of a physical host. This can be for a variety of reasons, such as a loss of power to the host or a motherboard failure.

When this event occurs, VMware HA will identify the failure of the host and will attempt to restart the protected virtual machines on a functional host.

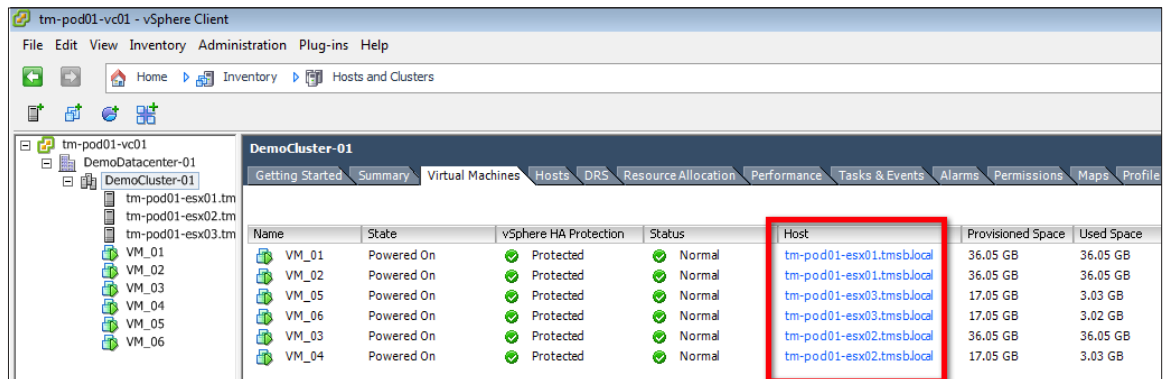


Figure 22. Checking Virtual Machines

First, use the vSphere Client to examine the virtual machines hosted within the cluster. In this example, we are going to cause the system tm-pod01-esx01.tmsb.local to fail. You need to check the virtual machines in your environment and ensure that at least one is online on the host that you are going to fail.

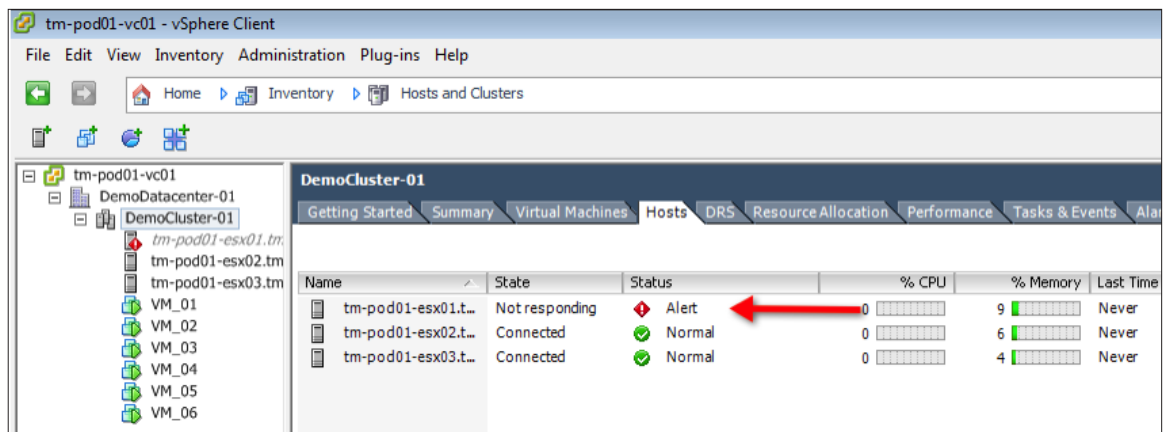


Figure 23. Removing Power from a Host

Next, remove the power from one of your hosts. By looking at the hosts within the cluster, you will see that VMware HA will detect the failure of the host and generate an alert.

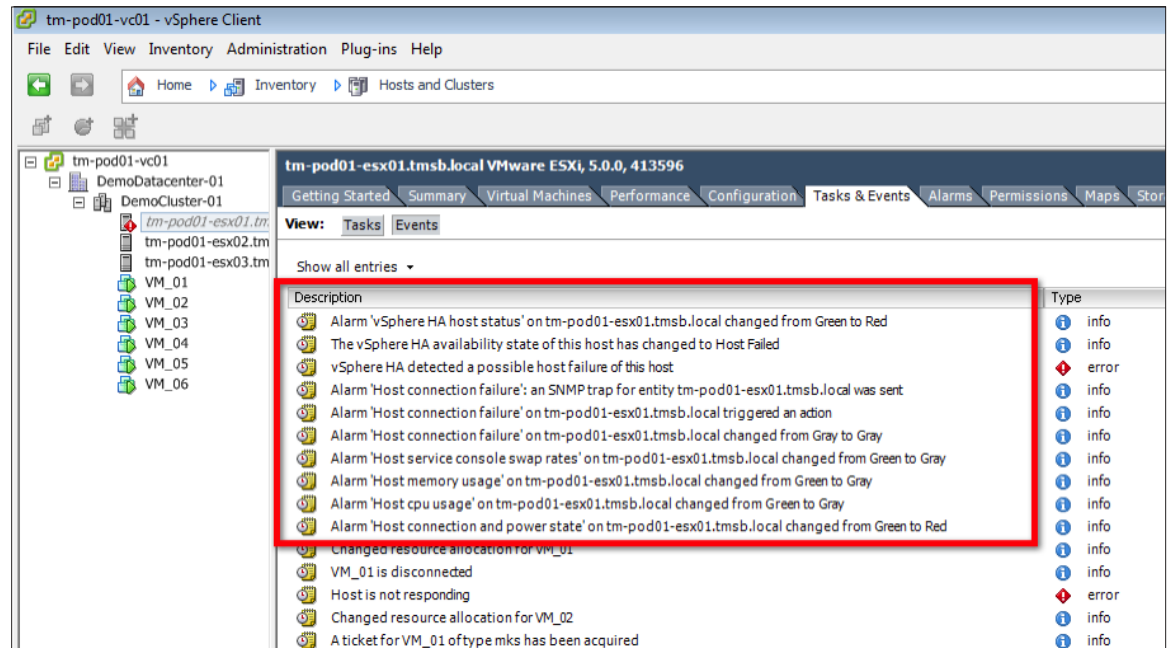


Figure 24. Failure Detection by VMware HA

By examining the events, you will see messages similar to the ones demonstrated in the preceding figure validating that VMware HA has detected the failure.

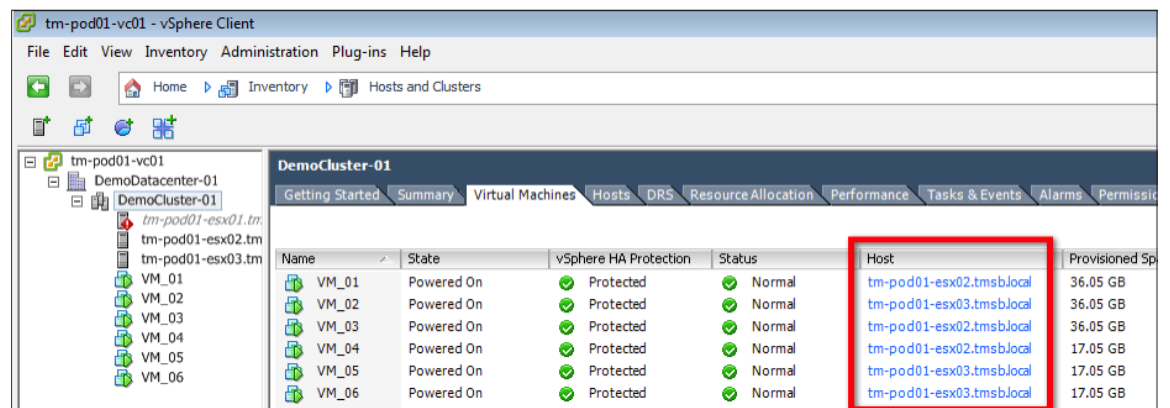


Figure 25. Virtual Machine View of a Cluster After Restart Attempt

After a failure of a host has been detected, HA will attempt to restart the virtual machines that were running on the failed host on other available hosts within the cluster. Go back to the virtual machine view of your cluster and notice that the virtual machines that were previously on the failed host are now online on other hosts.

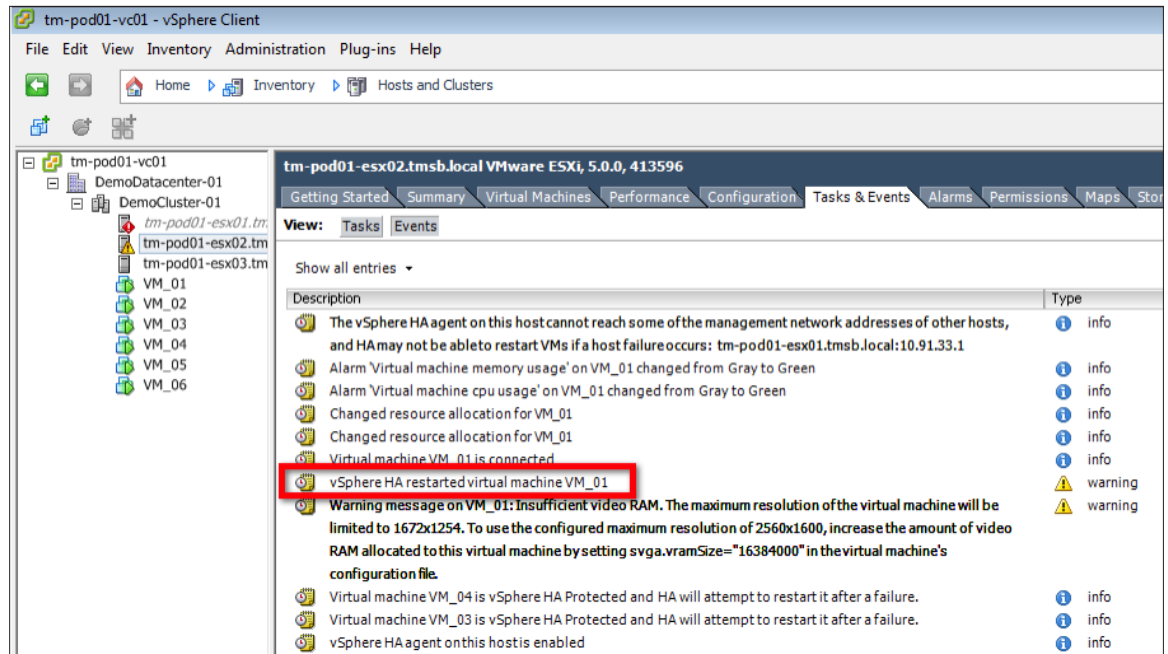


Figure 26. Viewing Log Messages After Restart Attempt

You can also examine the events for a host to see the log messages denoting that VMware HA has attempted to restart the virtual machine.

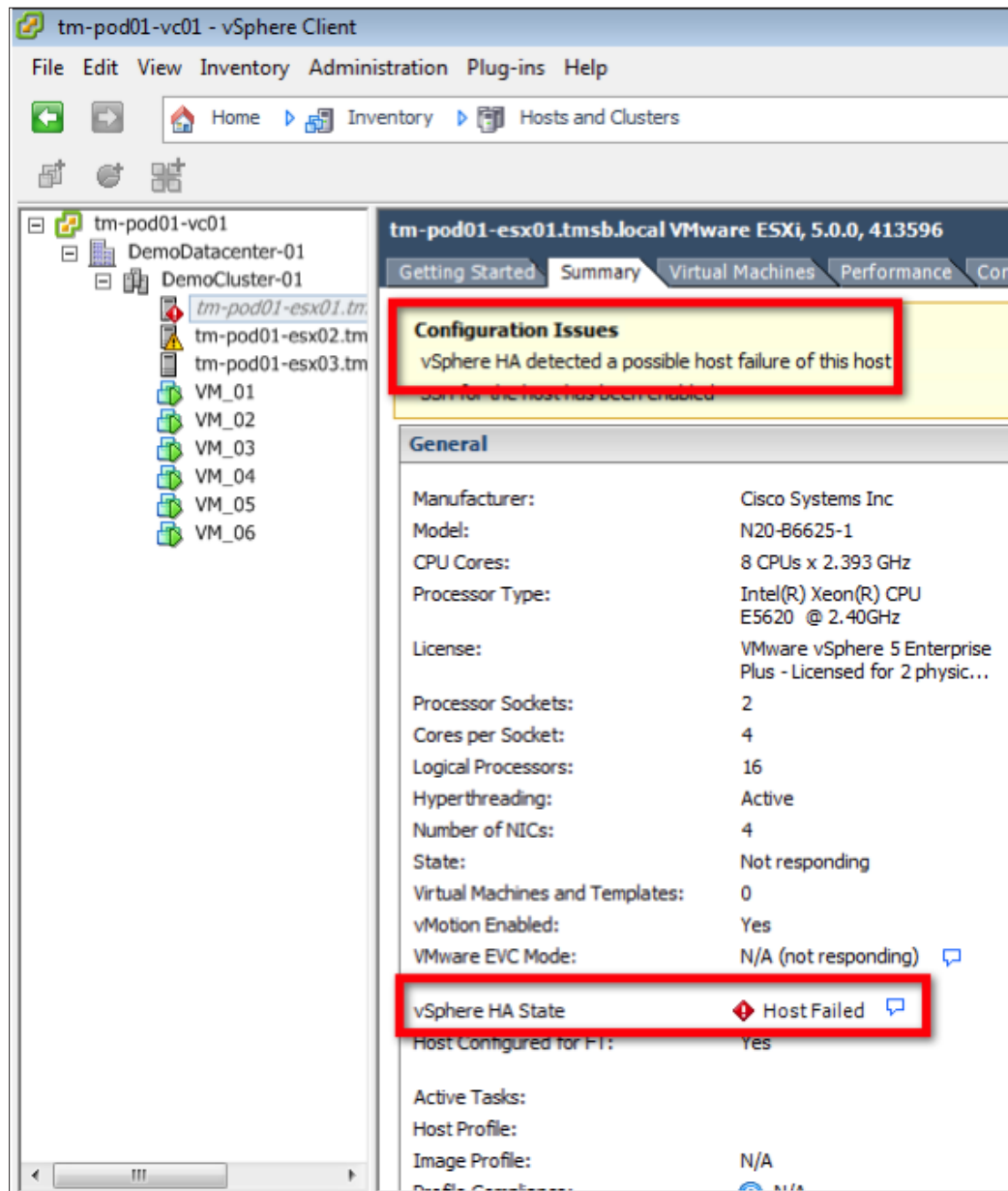


Figure 27. Summary of a Failed Host

By selecting the Summary tab for the failed host, you will notice that the issue is displayed in multiple places. The first is located at the top of the screen and second location is the vSphere HA State.

At this point, you will reapply power to the failed host and allow it to boot. Once it completes this process, you will see that it rejoins the cluster and continues to function as before.

Host Isolation

Host isolation occurs when a host loses the ability to communicate to other hosts within the cluster through the management network, and also loses the ability to ping the default isolation address. The following will demonstrate how to create this situation and induce the default actions that will be taken by VMware HA.

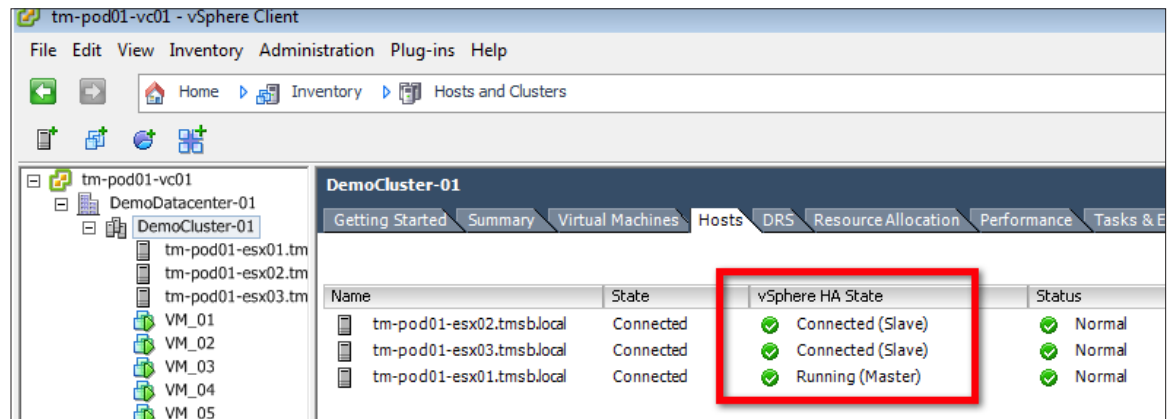


Figure 28. Identifying a Host to Be Isolated

First, you want to identify a host that will be isolated. For this example, host tm-pod01-esx03.tmsb.local has been chosen. You can verify that it is currently acting as a slave within the cluster.

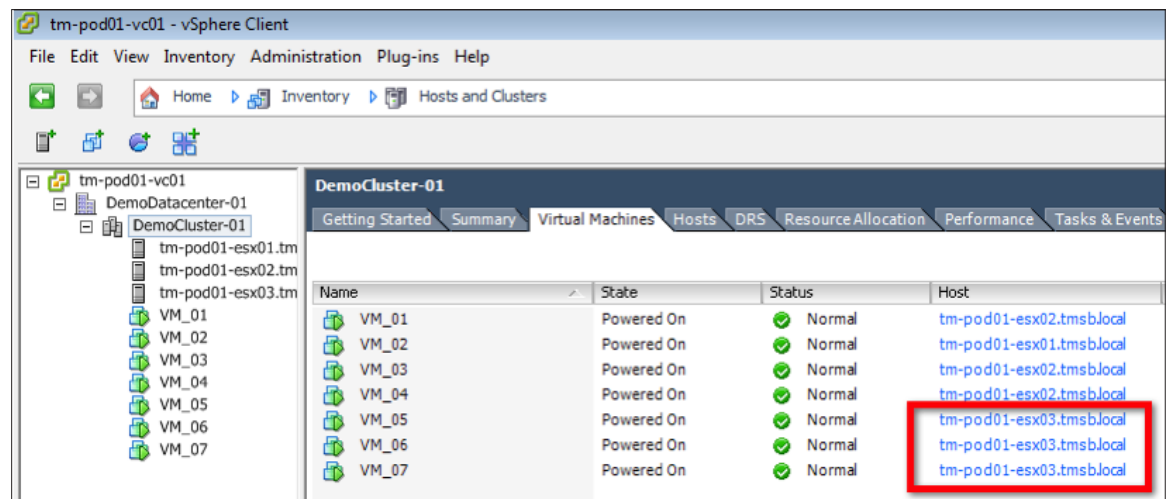


Figure 29. Identifying Virtual Machines on tm-pod01-esx03.tmsb.local

Now identify the virtual machines that are currently online on this host. These are the virtual machines that will be affected by the isolation response performed after the fault is inserted.

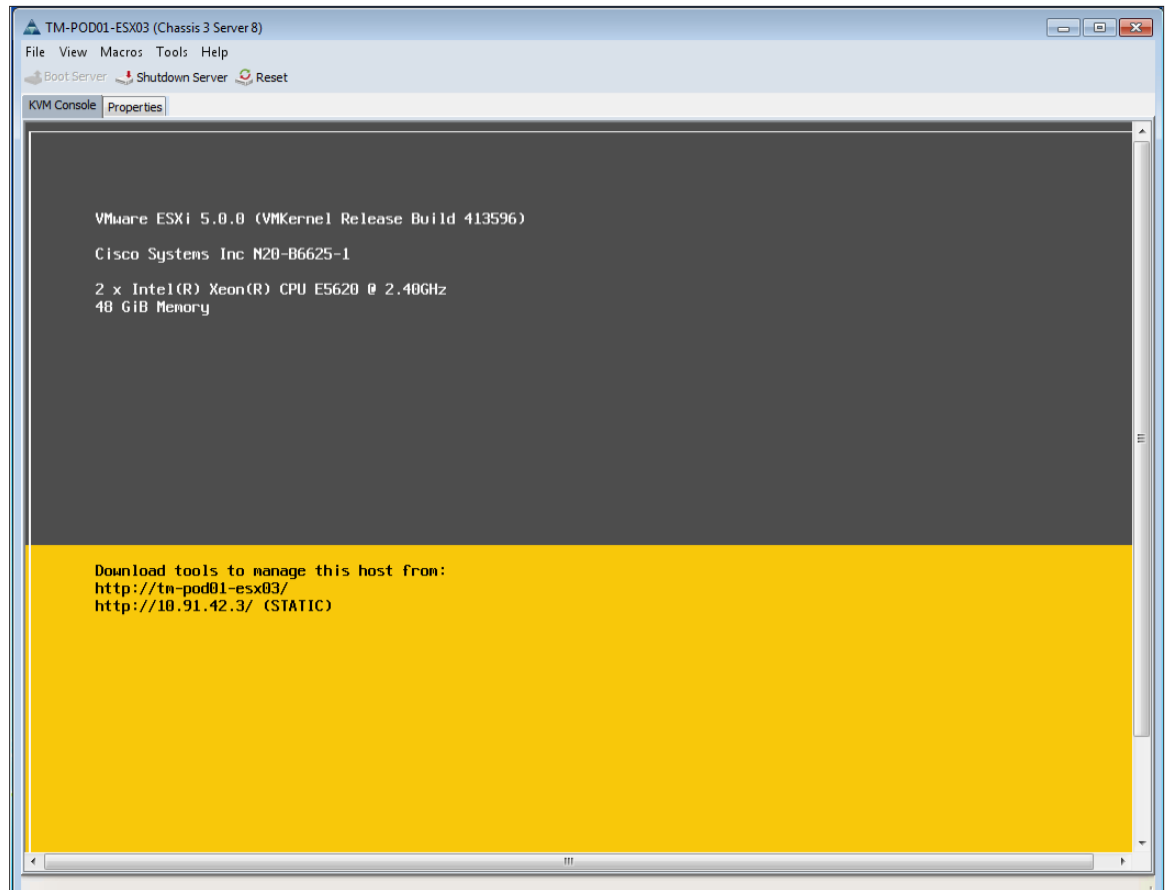


Figure 30. Obtaining Console Access to the Target Host

To insert a fault within the environment, you need to obtain console access to the target host. This will allow you to continue to access the host after the fault has been inserted, allowing you to recover gracefully afterwards. It is important to note that this procedure requires two networks – one for console access and another for those affected by the test. Refer to the “System Requirements” section for more information on the network configuration used.

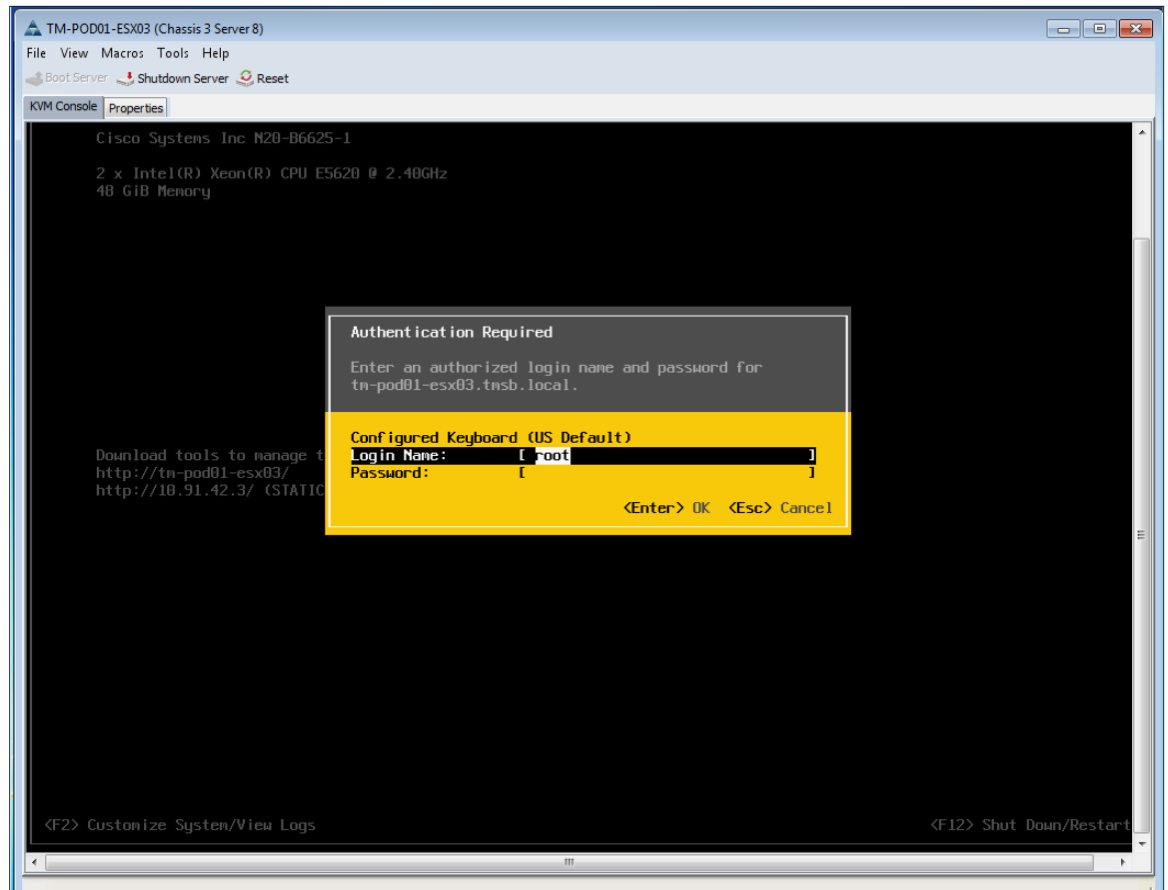


Figure 31. Authenticating to the Host

At the console, hit F2 to access the console menu. You will need to authenticate to the host first before it will allow access to the console menu.



Figure 32. Selecting Troubleshooting Options

Once you're logged in, select the **Troubleshooting Options** menu item.

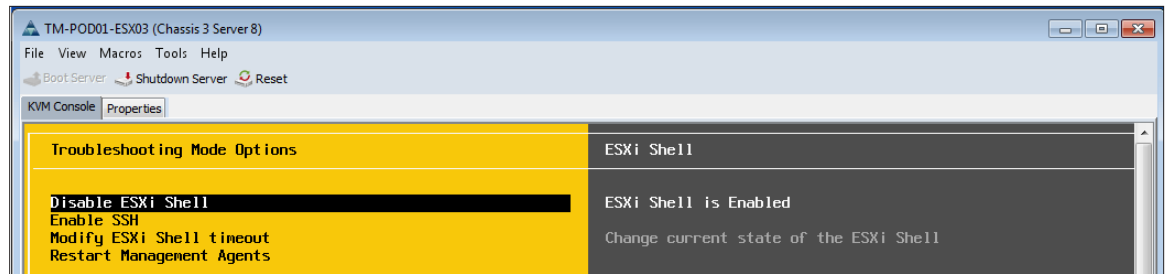


Figure 33. Enabling ESXi Shell

From here, select the **Enable ESXi Shell** option to enable the ESXi Shell. This shell will enable you to remove the network connections to the host.

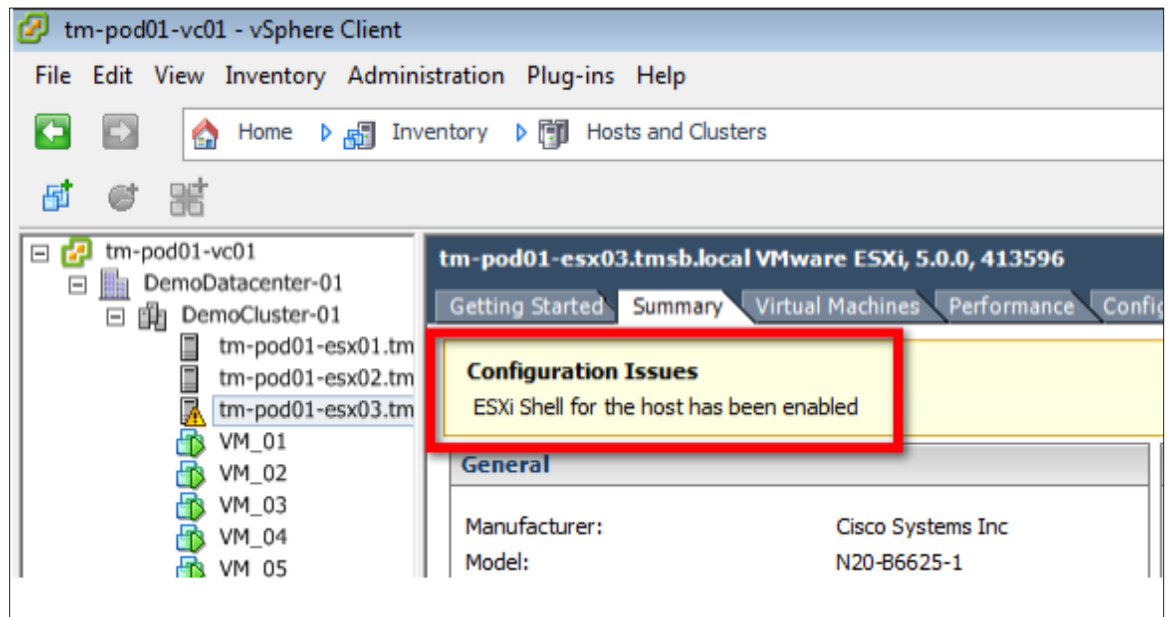


Figure 34. vSphere Client Warning Message

Once you do this, you will notice that the vSphere Client displays a warning message.

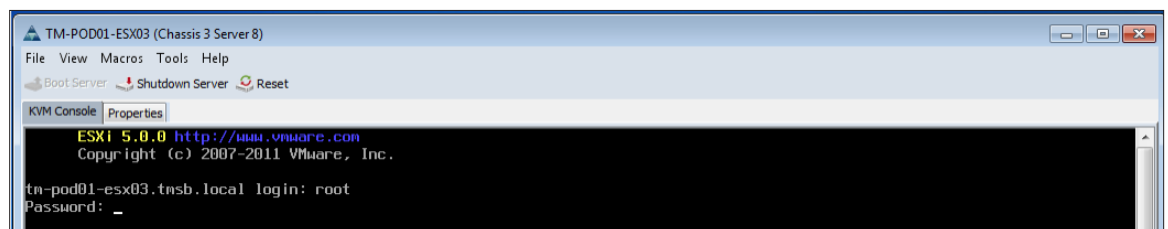


Figure 35. Accessing the ESXi Shell

At the ESXi console for the host, hit Alt-F1 to access the ESXi Shell. Log in to the shell using the user name and password specified for the host.

```

TM-POD01-ESX03 (Chassis 3 Server 8)
File View Macros Tools Help
Boot Server Shutdown Server Reset
KVM Console Properties
ESXi 5.0.0 http://www.vmware.com
Copyright (c) 2007-2011 VMware, Inc.

tm-pod01-esx03.tnsb.local login: root
Password:
The time and date of this login have been sent to the system logs.

VMware offers supported, powerful system administration tools. Please
see www.vmware.com/go/sysadmintools for details.

The ESXi Shell can be disabled by an administrative user. See the
vSphere Security documentation for more information.
# esxcfg-vswitch -l
Switch Name      Num Ports  Used Ports  Configured Ports  MTU      Uplinks
vSwitch0         128        11         128              1500     vmnic0,vmnic1

  PortGroup Name  VLAN ID  Used Ports  Uplinks
  Production02    3001     3           vmnic0,vmnic1
  Management Network 2912     1           vmnic0,vmnic1
  vMotion01       3002     1           vmnic0,vmnic1
  FT01            3003     1           vmnic0,vmnic1
  iSCSI01         3004     1           vmnic0,vmnic1
  HBR01           3005     1           vmnic0,vmnic1

~ # esxcfg-vswitch -U vmnic1 vSwitch0
~ # esxcfg-vswitch -U vmnic0 vSwitch0
~ # esxcfg-vswitch -l
Switch Name      Num Ports  Used Ports  Configured Ports  MTU      Uplinks
vSwitch0         128        9          128              1500

  PortGroup Name  VLAN ID  Used Ports  Uplinks
  Production02    3001     3           vmnic0
  Management Network 2912     1           vmnic0
  vMotion01       3002     1           vmnic0
  FT01            3003     1           vmnic0
  iSCSI01         3004     1           vmnic0
  HBR01           3005     1           vmnic0

~ # _
  
```

Figure 36. Using the esxcfg-vswitch Command

In order to disrupt the network connection to the host, you can use the esxcfg-vswitch command. Using esxcfg-vswitch -l, obtain a list of the uplinks that are present on the host. In this example, there are two – vmnic0 and vmnic1 – that can be identified on vSwitch0.

Use the command esxcfg-vswitch -U <uplink> <switch>, where uplink is an identified uplink and switch is the name of the switch the uplink is connected to, in order to remove the uplinks from the virtual switch. Ensure that you do this for all of the uplinks previously identified. Once completed, verify that all of the uplinks have been removed by using the esxcfg-vswitch -l command again.

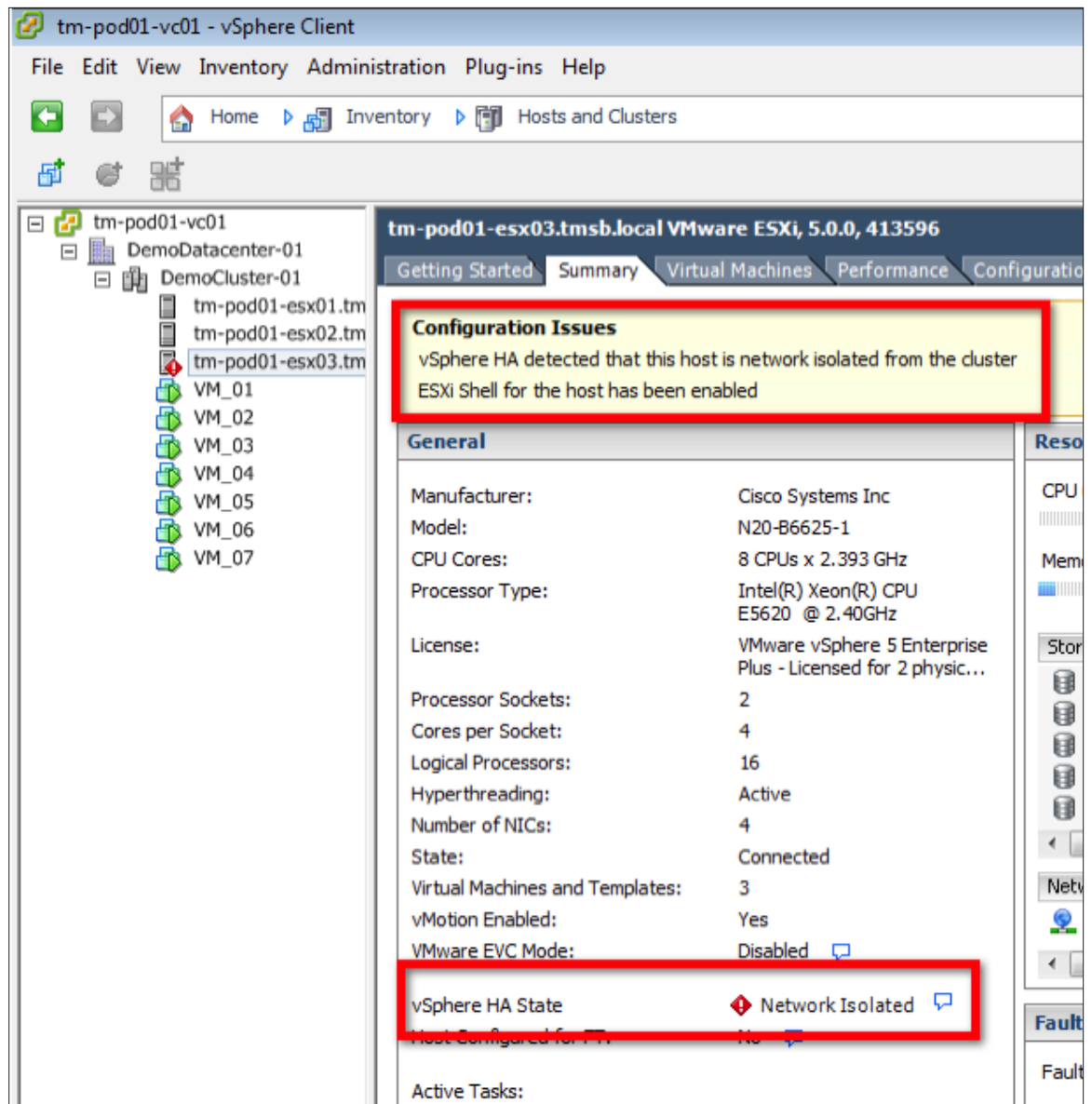


Figure 37. Identifying Host Isolation

Using the vSphere Client, select the host from the left-hand navigation pane and select the Summary tab. The host will be identified as being isolated both at the top of the screen and in the vSphere HA State notification.

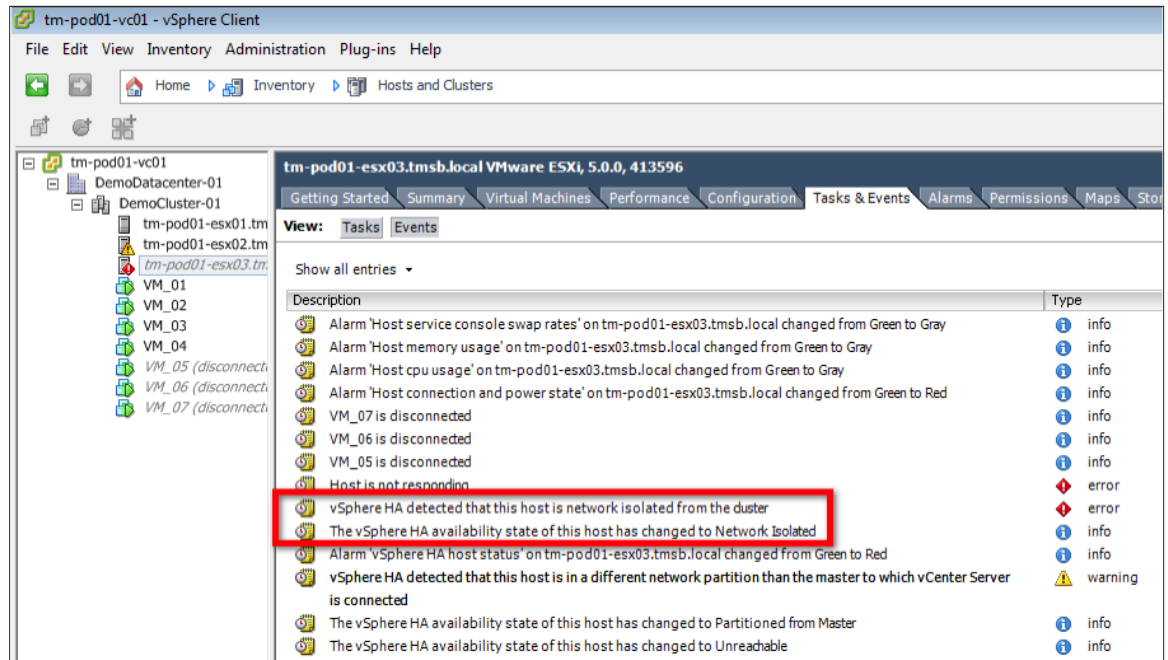


Figure 38. Log Messages About Host Isolation

Moving to the Tasks & Events tab, you will also see the log messages that were generated when VMware HA detected the host isolation.

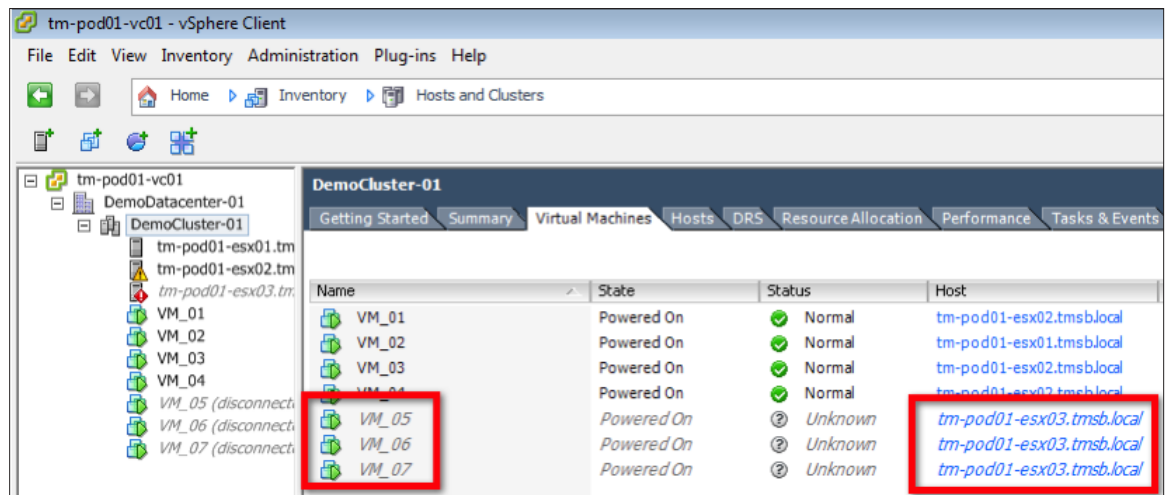


Figure 39. Observing the Virtual Machines on the Isolated Host

Examine the output of the Virtual Machines tab of the cluster. Observe that the virtual machines on the isolated host are now shown in gray. You'll also observe that the virtual machines did not get restarted on another host. This is due to the fact that the default setting for the isolation response is **Leave Powered On**. With this as the setting for the isolation response, the virtual machines will continue to run on the isolated host. In this scenario, setting the isolation response to **Shutdown** would cause the virtual machines to gracefully shut down, then restart.

If you would like to see the effects of the various isolation response settings in this situation, simply change the isolation response to the desired setting and perform this test again.

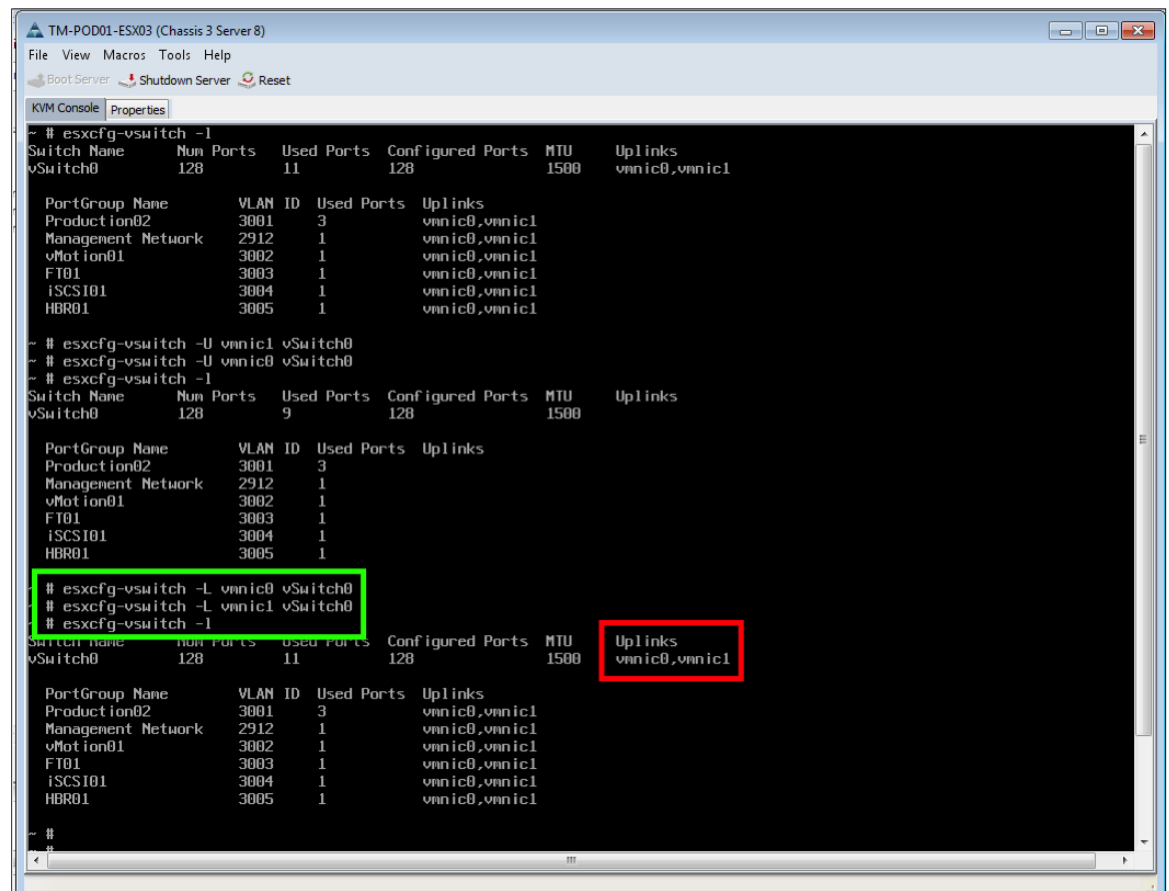


Figure 40. Restoring Uplinks for the Host with the esxcfg-vswitch -l Command

To restore normal operation, utilize the ESXi Shell to execute the `esxcfg-vswitch -l` command for each of the uplinks that were previously removed. Use the `esxcfg-vswitch -l` command to verify that the uplinks have been restored.

Log out of the ESXi Shell by typing **exit** at the prompt. Use Alt-F1 to return to the console screen.

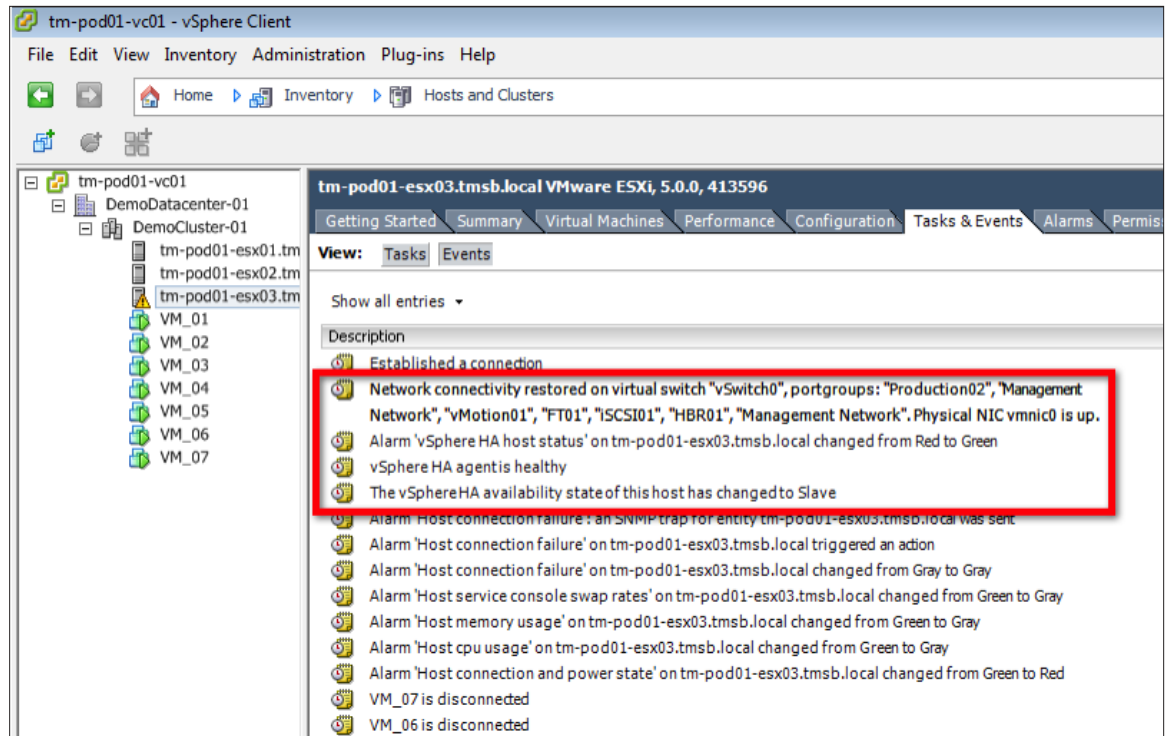


Figure 41. Examining Events for the Host

Once you restore the uplinks for the host, you can utilize the vSphere Client to examine the events for the host. This will show you that communication with the other hosts in the cluster has been re-established.

Even after you have re-established the network connections, you'll notice that the host still displays a warning. This warning is due to the fact that the ESXi Shell is still enabled.

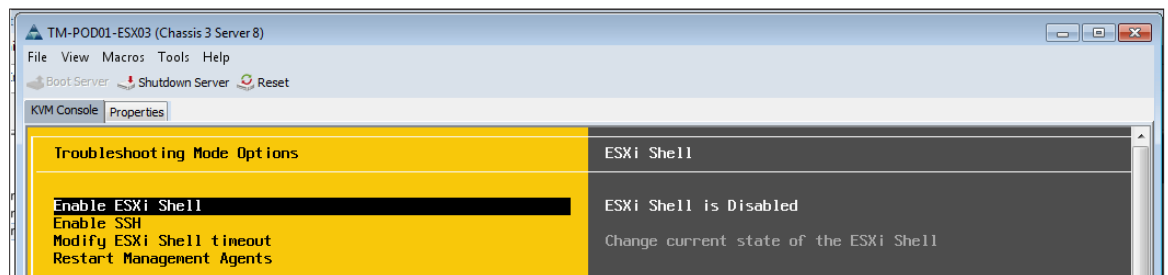


Figure 42. Disabling ESXi Shell Access

Using the console, select **Disable ESXi Shell** under the Troubleshooting Mode Options screen to disable ESXi Shell access.

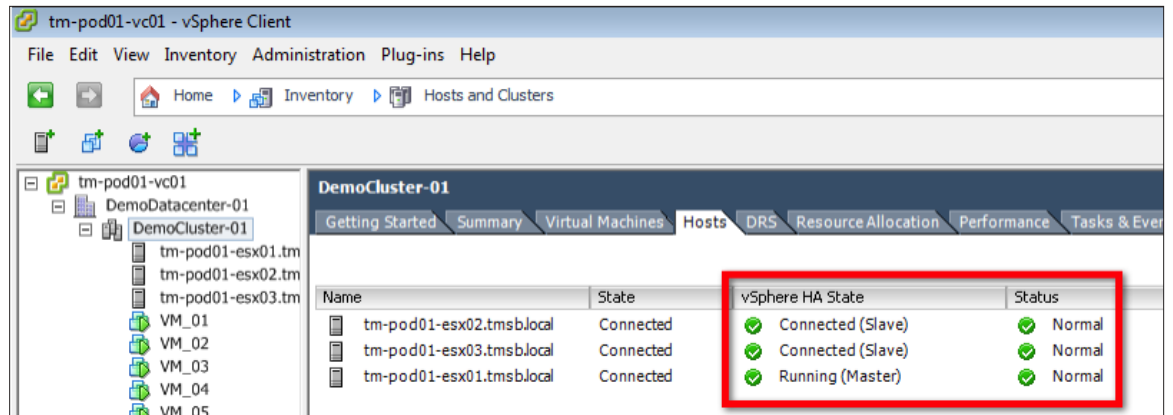


Figure 43. Verifying That Host is Operating Normally

Use the vSphere Client to show all of the hosts within the cluster and verify that the previously isolated host is now operating normally and has reconnected to the cluster.

Disabling VMware HA

As with the enabling of HA, disabling HA is a simple, straightforward process. This section will walk you through the required steps before continuing on to the next topic.

Connect to a Virtual Server



Figure 44. Connecting to a Virtual Server Instance

Using the vSphere Client, connect to your virtual server instance.

Go to the Cluster Summary

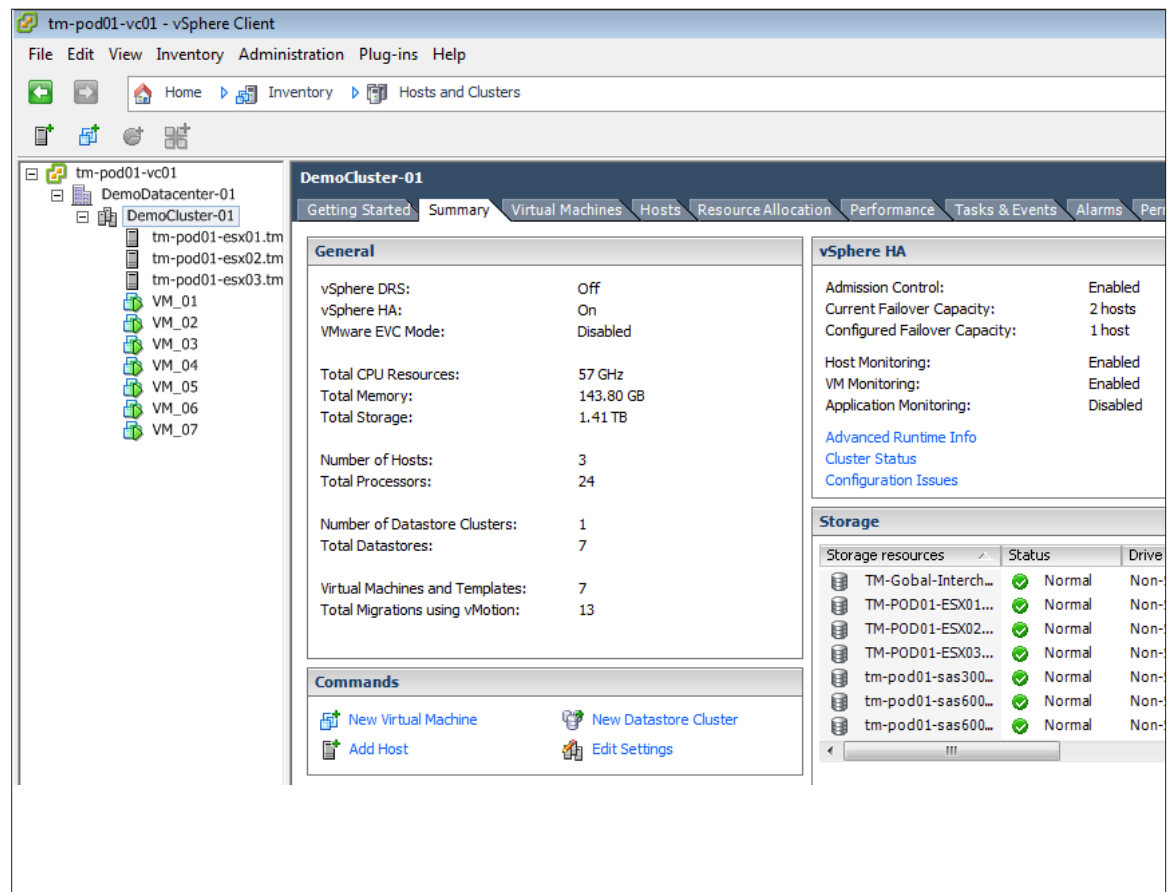


Figure 45. Cluster Summary Screen

Once connected to your virtual server instance, select your cluster by clicking its name on the left-hand panel. Select the **Summary** tab to bring up the cluster summary screen.

Edit Cluster Settings

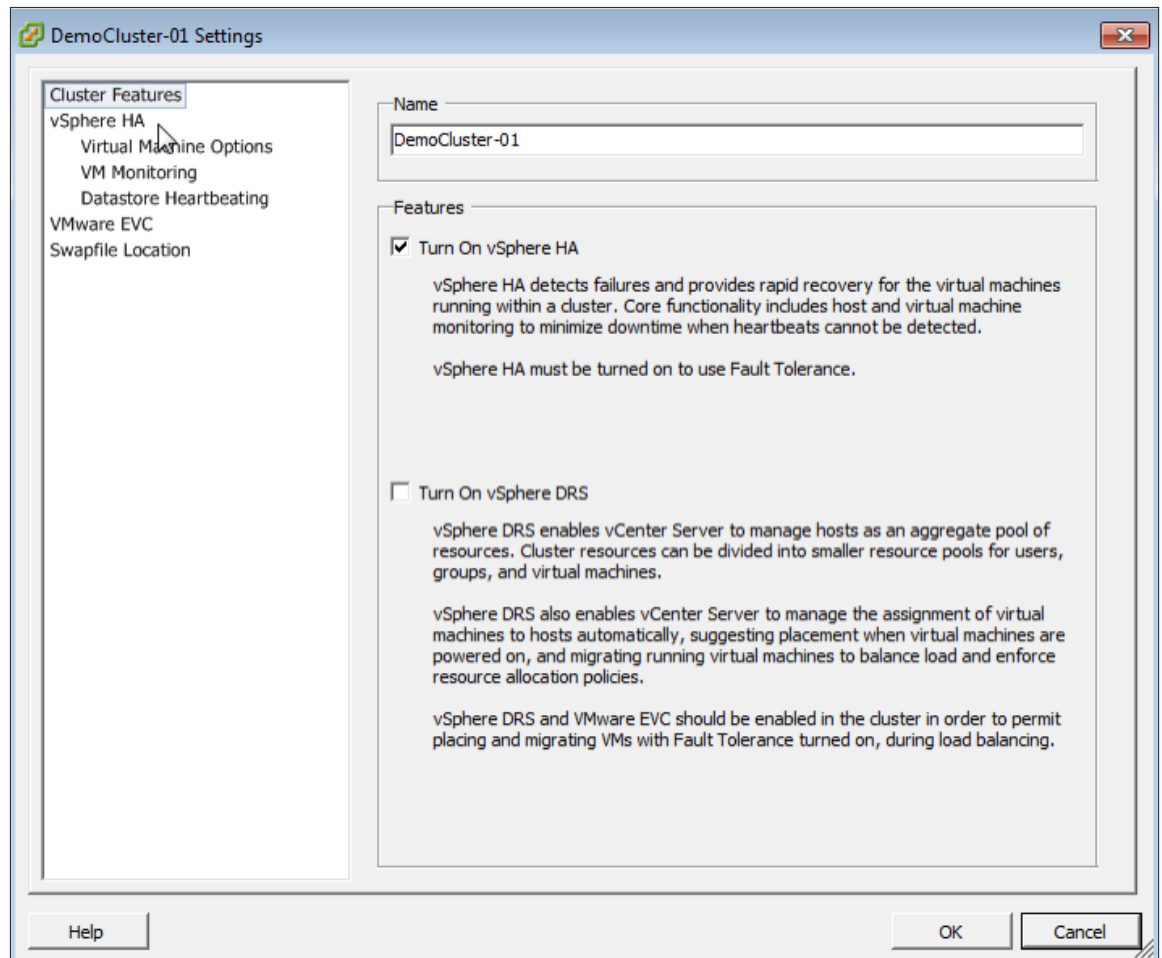


Figure 46. Cluster Settings Wizard

In the cluster summary screen, select the **Edit Settings** option. This will bring up a wizard that you can use to modify the settings of the cluster. Click the check box next to **Turn On vSphere HA** to deselect it and select OK. This will close the wizard and the system will unconfigure VMware HA.

Recent Tasks		
Name	Target	Status
Unconfiguring vSphere HA	tm-pod01-esx02.tmsb.local	In Progress
Unconfiguring vSphere HA	tm-pod01-esx03.tmsb.local	In Progress
Unconfiguring vSphere HA	tm-pod01-esx01.tmsb.local	In Progress

Figure 47. Viewing the Progress of the Unconfigure Task of VMware HA

Under the Recent Tasks pane of the vSphere Client, you can observe the progress of the unconfigure task of HA on the systems within the cluster.

Wait for Task to Complete










Recent Tasks		
Name	Target	Status
 Unconfiguring vSphere HA	 tm-pod01-esx02.tmsb.local	 Completed
 Unconfiguring vSphere HA	 tm-pod01-esx03.tmsb.local	 Completed
 Unconfiguring vSphere HA	 tm-pod01-esx01.tmsb.local	 Completed

Figure 48. Unconfigure Tasks Completed

Wait until all the unconfigure tasks show a Completed status.

Getting Familiar with the New Command-Line Interface

Introduction

vSphere supports several command-line interfaces for managing your virtual infrastructure, including the VMware vSphere® Command-Line Interface (vCLI), a set of ESXi Shell commands, and VMware vSphere® PowerCLI. You can choose the CLI set best suited to your needs. The following table provides a summary of the command-line tools available in vSphere 5.0.

COMMANDS	STATE IN 5.0	AVAILABILITY
esxcli commands	New in vSphere 5.0	<ul style="list-style-type: none"> Available from the ESXi Shell and the vCLI Used for local and remote administration Used to manage most aspects of vSphere
vicfg- commands	Minor changes in vSphere 5.0	<ul style="list-style-type: none"> Available from the vCLI only Used for remote administration only Augments the esxcli commands to manage aspects not yet covered by esxcli
Other commands (vmware-cmd, vifs)	Minor changes in vSphere 5.0	<ul style="list-style-type: none"> Available from the vCLI only Used for remote administration only Additional Perl commands used to manage aspects not covered with esxcli or vicfg-
vSphere PowerCLI	Minor changes and updates in vSphere 5.0	<ul style="list-style-type: none"> vSphere PowerCLI Used for remote administration only Used to administer ESXi hosts from Windows systems

Table 1. Summary of vSphere 5.0 Command-Line Tools

This section of the *VMware vSphere 5.0 Evaluation Guide, Volume One*, covers the new esxcli command-line interface. The esxcli command allows you to manage many aspects of an ESXi host. You can run esxcli commands remotely from the vCLI or locally from the ESXi Shell.

NOTE: The ESXi Shell is intended for advanced users, because even minor mistakes in the shell can result in serious problems. Users should use the vCLI for routine CLI administration and only fall back on the ESXi Shell when necessary. It is recommended that use of the ESXi Shell be limited to situations when you are working under the direction of the VMware Technical Support staff.

The New esxcli Command

For the first time, the new esxcli command is unified for both local and remote command-line administration. In addition, the esxcli command has been enhanced to perform many tasks previously only performed with the **vicfg-** commands. However, it does not yet perform all the tasks. When performing configuration tasks from the command line, the esxcli command is the preferred command. Only fall back to the **vicfg-** and other vCLI commands when there is no esxcli command available. Moving forward, all the vCLI commands are scheduled to be replaced by esxcli commands.

NOTE: In vSphere 5.0, the esxcli command does not yet provide a full set of command capabilities. Continue to use the esxcli command in conjunction with the vicfg- and other vCLI commands (that is, vmware-cmd, vmkfstools, and vifs). The esxcli command in vSphere 5.0 is not backward compatible with earlier versions of the command, because it introduces a new syntax that is different from earlier vSphere releases.

esxcli Command-Line Syntax

The esxcli command is made up of a hierarchy of namespaces. At each level of the hierarchy there are additional namespaces and commands. This provides for a user-friendly CLI interface that allows for the easy discovery of the command syntax.

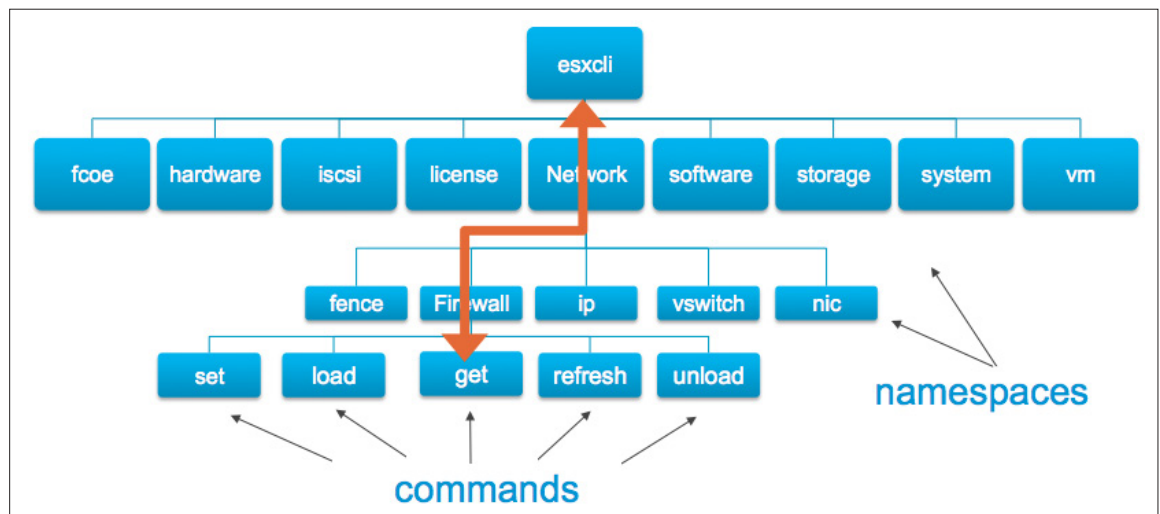


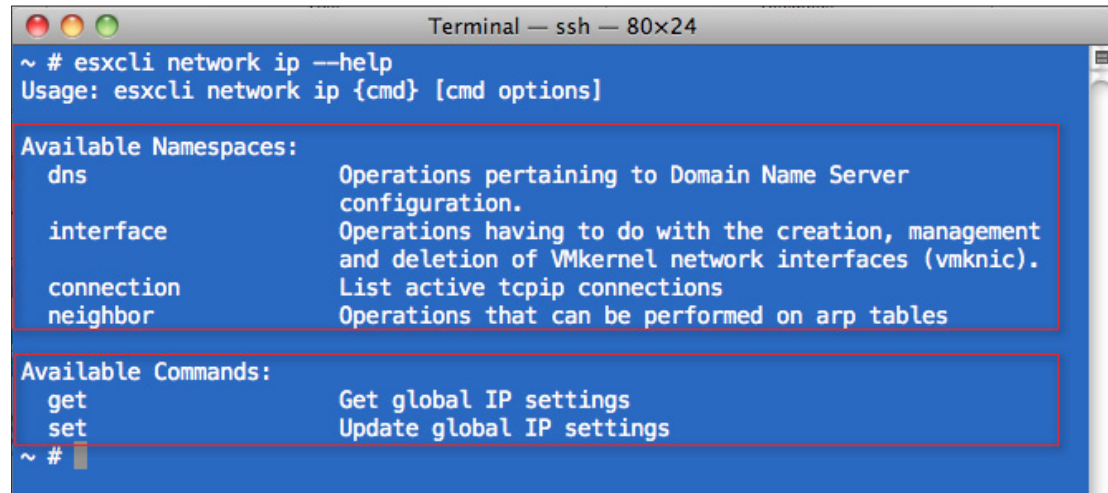
Figure 49. esxcli Namespace Hierarchy

The preceding figure provides a graphic illustration of the command to query the ESXi firewall. The user invokes the **esxcli** command with the **network** namespace, the **firewall** sub-namespace, and the **get** command. The following is an example of this command:

```
Terminal — ssh — 80x24
~ # esxcli network firewall get
Default Action: DROP
Enabled: true
Loaded: true
~ #
```

Figure 50. esxcli network firewall get Command

At any time, you can use the **--help** option to discover information about the available namespaces and commands relative to your current namespace. In the following example, the **--help** parameter is used to get more information about the available namespaces and commands under the **network** namespace:



```

Terminal — ssh — 80x24
~ # esxcli network ip --help
Usage: esxcli network ip {cmd} [cmd options]

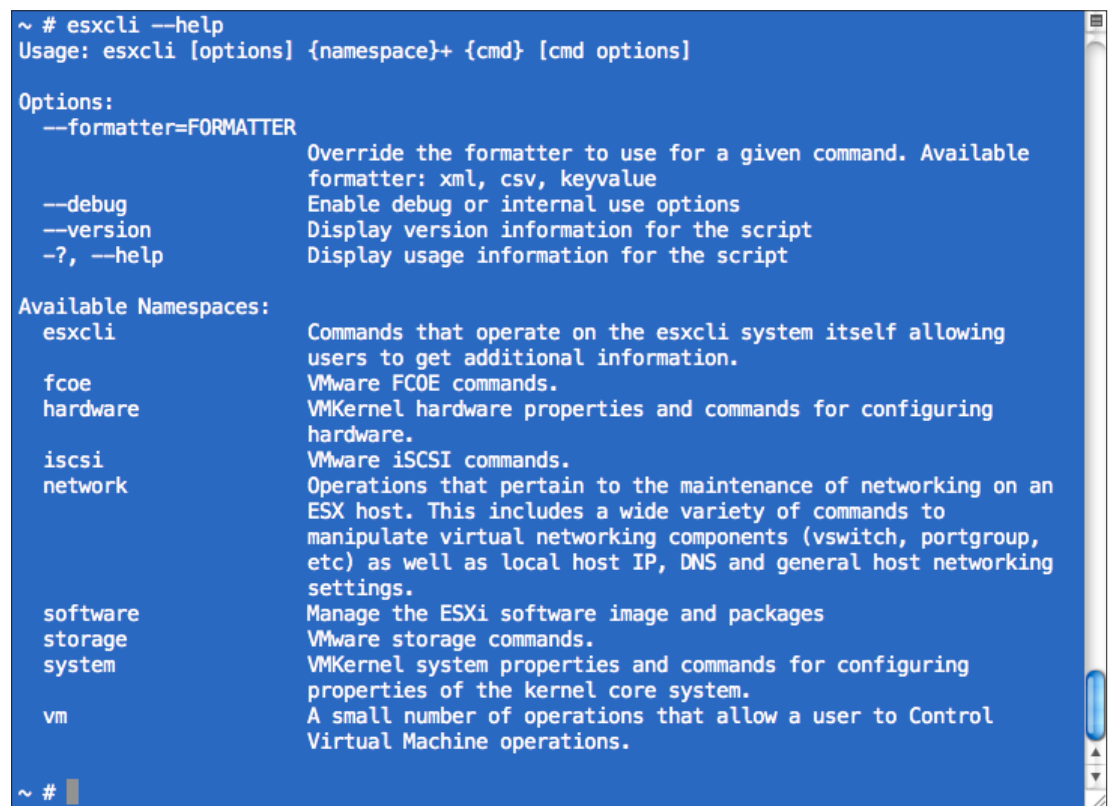
Available Namespaces:
  dns           Operations pertaining to Domain Name Server
                configuration.
  interface     Operations having to do with the creation, management
                and deletion of VMkernel network interfaces (vmknic).
  connection    List active tcpip connections
  neighbor      Operations that can be performed on arp tables

Available Commands:
  get           Get global IP settings
  set           Update global IP settings
~ #

```

Figure 51. Using the --help Parameter to Get Information About Namespaces and Commands

Every **esxcli** command is comprised of the **esxcli** command followed, if needed, by one or more **options**, followed by one or more **namespaces**, followed by the **command** to be executed along with any **command options**. The following screen shot shows the **esxcli** usage screen:



```

~ # esxcli --help
Usage: esxcli [options] {namespace}+ {cmd} [cmd options]

Options:
  --formatter=FORMATTER  Override the formatter to use for a given command. Available
                        formatter: xml, csv, keyvalue
  --debug                Enable debug or internal use options
  --version              Display version information for the script
  -?, --help             Display usage information for the script

Available Namespaces:
  esxcli                 Commands that operate on the esxcli system itself allowing
                        users to get additional information.
  fcoe                   VMware FCoE commands.
  hardware               VMkernel hardware properties and commands for configuring
                        hardware.
  iscsi                  VMware iSCSI commands.
  network                Operations that pertain to the maintenance of networking on an
                        ESX host. This includes a wide variety of commands to
                        manipulate virtual networking components (vswitch, portgroup,
                        etc) as well as local host IP, DNS and general host networking
                        settings.
  software               Manage the ESXi software image and packages
  storage                VMware storage commands.
  system                 VMkernel system properties and commands for configuring
                        properties of the kernel core system.
  vm                     A small number of operations that allow a user to Control
                        Virtual Machine operations.

~ #

```

Figure 52. esxcli -help Example

Although the `esxcli` command is unified for both local and remote administration, the syntax does vary slightly, depending upon if you are running commands locally from the ESXi Shell or remotely through the vCLI.

- When running `esxcli` commands locally from the ESXi Shell, the target host is always the local host on which the command is run. In addition, the login credentials are always assumed to be those of the logged-in user.
- When running the `esxcli` commands remotely, you must specify the target ESXi host (or VMware vCenter Server™), along with the user credentials used to execute the command.

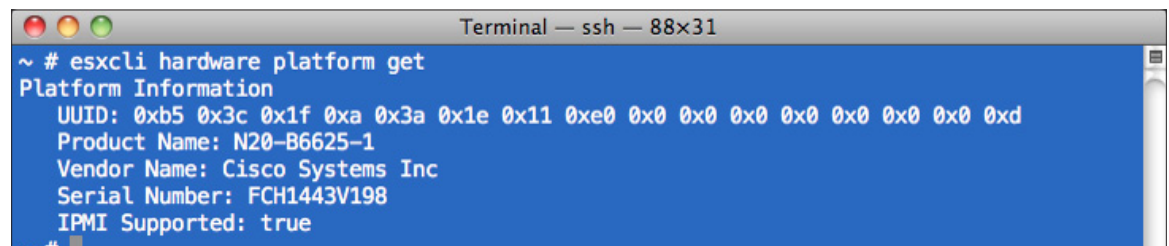
Remote `esxcli` Command Authentication

When running `esxcli` commands remotely, you must specify a target ESXi host or vCenter server and provide the user credentials for the command. The following are methods available to perform user authentication:

- Using command-line options
- Using a session file
- Using environment variables
- Using a configuration file
- Using Microsoft Windows `--passthroughauth`
- Using VMware vSphere® Management Assistant (vMA) `vi-fastpass`

Details for each method are documented in the *Getting Started with vSphere Command-Line Interfaces* guide.

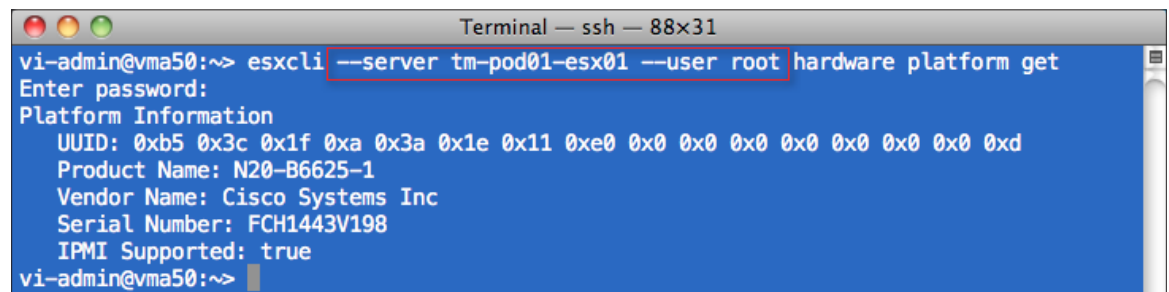
The following examples illustrate the different syntax required when running the **`esxcli hardware platform get`** command from the local ESXi Shell, compared to running it remotely from the vCLI. You must add the `--server` and `--user` options when running the command remotely, in addition to being prompted to enter the password.



```

Terminal — ssh — 88x31
~ # esxcli hardware platform get
Platform Information
  UUID: 0xb5 0x3c 0x1f 0xa 0x3a 0x1e 0x11 0xe0 0x0 0x0 0x0 0x0 0x0 0x0 0xd
  Product Name: N20-B6625-1
  Vendor Name: Cisco Systems Inc
  Serial Number: FCH1443V198
  IPMI Supported: true
  
```

Figure 53. Sample `esxcli` Command Run from ESXi Shell



```

Terminal — ssh — 88x31
vi-admin@vma50:~> esxcli --server tm-pod01-esx01 --user root hardware platform get
Enter password:
Platform Information
  UUID: 0xb5 0x3c 0x1f 0xa 0x3a 0x1e 0x11 0xe0 0x0 0x0 0x0 0x0 0x0 0x0 0xd
  Product Name: N20-B6625-1
  Vendor Name: Cisco Systems Inc
  Serial Number: FCH1443V198
  IPMI Supported: true
vi-admin@vma50:~>
  
```

Figure 54. Sample `esxcli` Command Run Remotely from vMA

Enabling Access to the ESXi Shell

Before you can run esxcli commands on the host, you must enable the ESXi Shell. Complete the steps in this section to enable the ESXi Shell on each ESXi host.

Enabling the ESXi Shell from the DCUI

Perform the following steps in order to enable the ESXi Shell while logged into the DCUI:

- Log in to the ESXi host DCUI.
- Select **Troubleshooting Mode Options**.
- Select **Enable ESXi Shell** and press return.
- Press Alt-F1 to access the ESXi Shell.

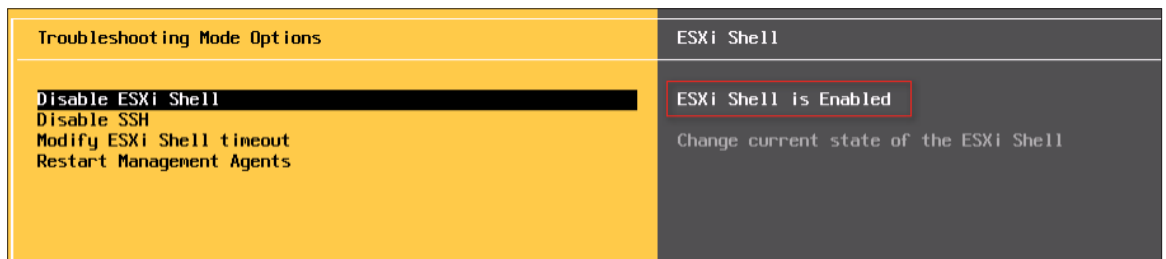


Figure 55. Enable ESXi Shell from DCUI

Enabling the ESXi Shell from the vSphere Client

Perform the following steps to enable the ESXi Shell while logged into the vSphere Client:

- Log in to the vSphere Client.
- Select the ESXi host and choose Configuration -> Security Profile.
- From the **Services** section, select **Properties**.
- Select the **ESXi Shell** option and choose **Options**.
- Select **Start** to start the ESXi Shell, enabling local access to the ESXi Shell.

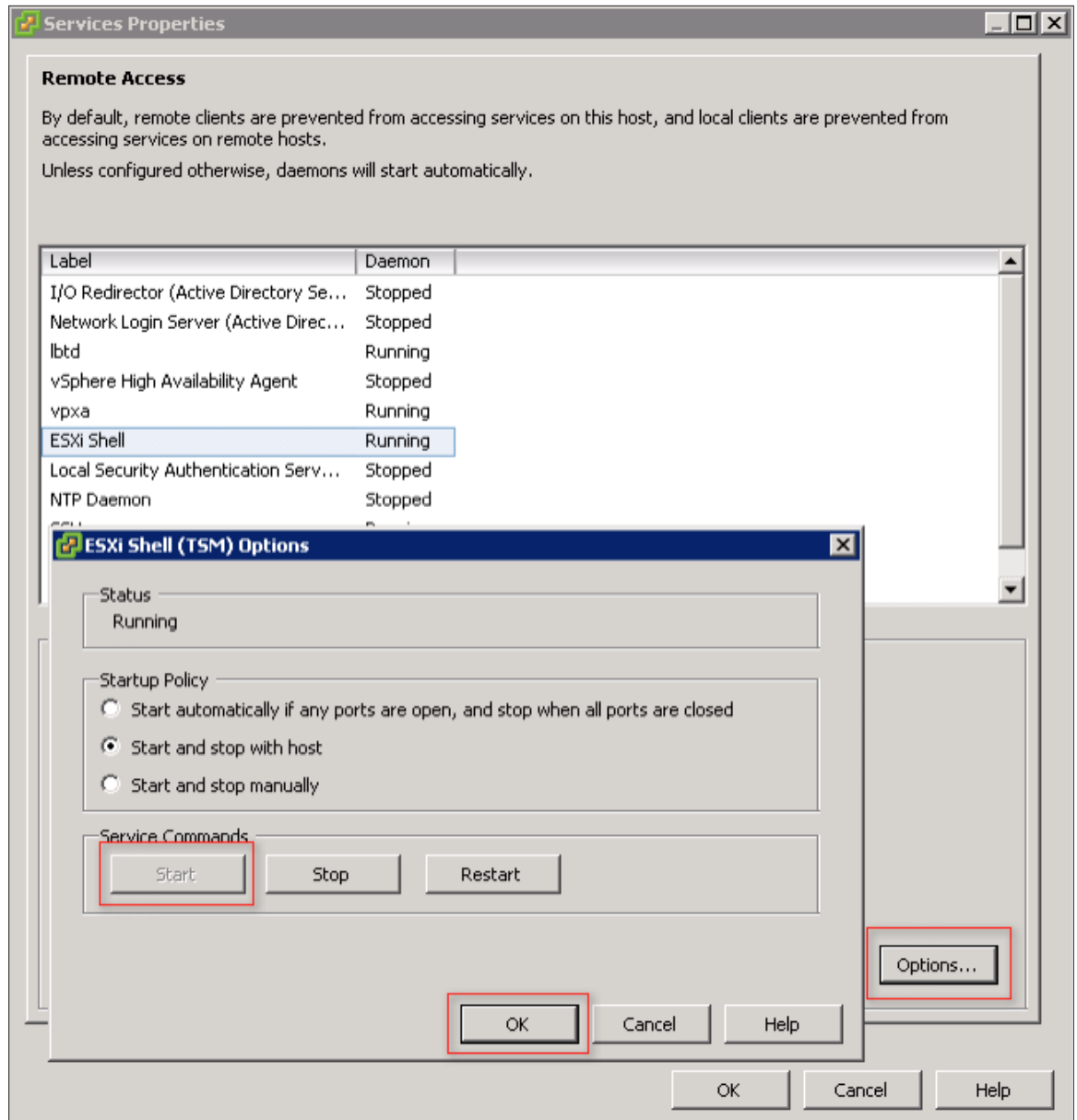


Figure 56. Enable ESXi Shell from vSphere Client

Enabling SSH Access to the ESXi Shell

In addition to running commands directly from the ESXi console, you can also enable SSH services to allow remote access to the ESXi Shell. The following section shows how to enable SSH access to the ESXi Shell.

Enabling SSH from the DCUI

Perform the following steps to enable the ESXi Shell from the DCUI:

- Log in to the ESXi host DCUI.
- Select **Troubleshooting Options**.
- Select **Enable ESXi Shell** and press return.
- Press Alt-F1 to access the ESXi Shell.

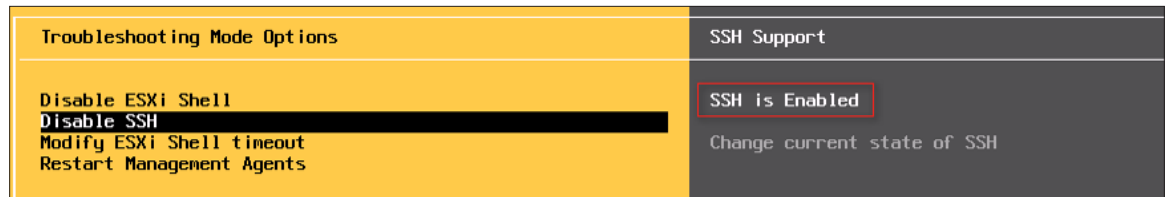


Figure 57. Enable SSH from DCUI

Enabling the SSH from the vSphere Client

Perform the following steps to enable the ESXi Shell from the vSphere Client:

- Log in to the vSphere Client.
- Select the ESXi host and choose Configuration -> Security Profile.
- From the **Services** section, select **Properties**.
- Select the **SSH** option and choose **Options**.
- Select **Start** to start SSH on the host.

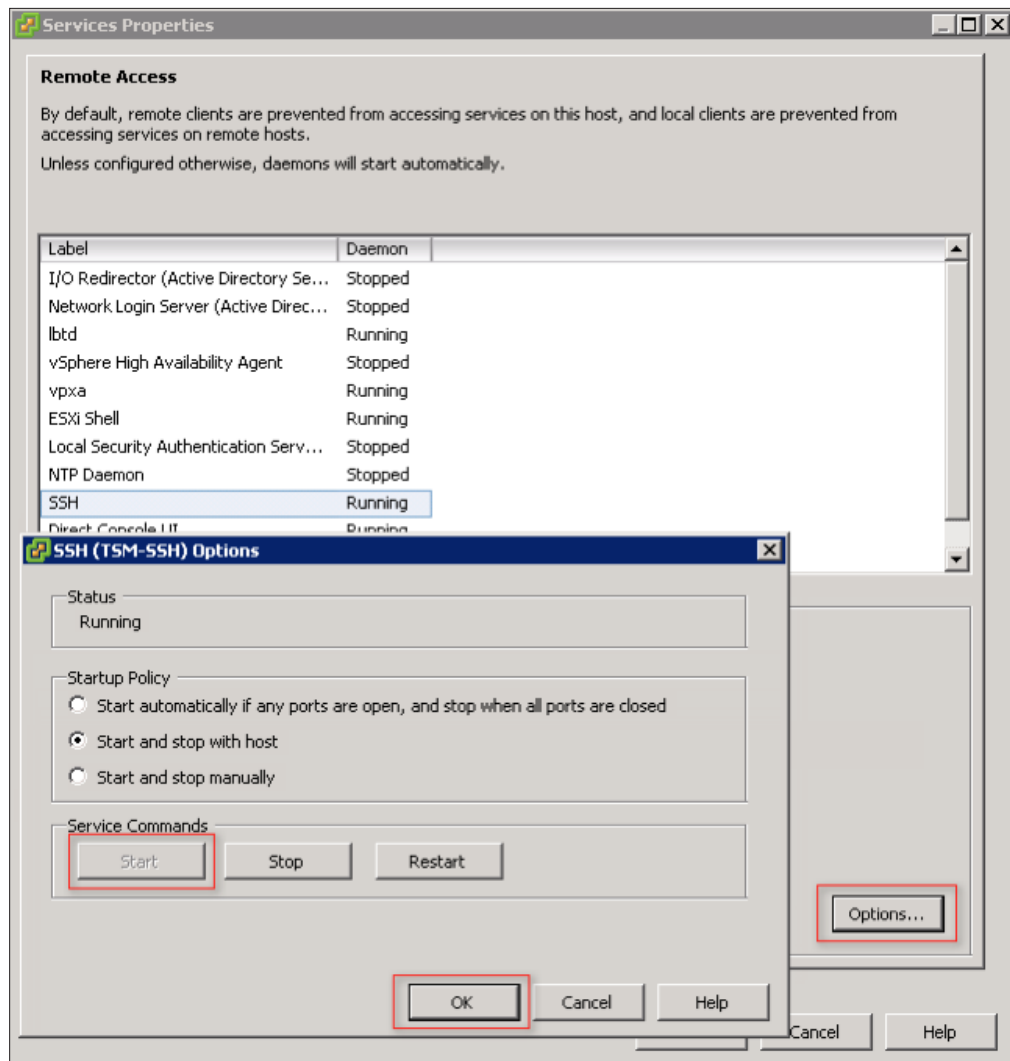


Figure 58. Enable SSH from vSphere Client

vSphere Client Notification When the ESXi Shell and SSH Are Enabled

Any time the ESXi Shell or SSH is enabled on a host, the vSphere Client will show a warning on the host summary page serving as a reminder to disable the access when it is no longer needed.

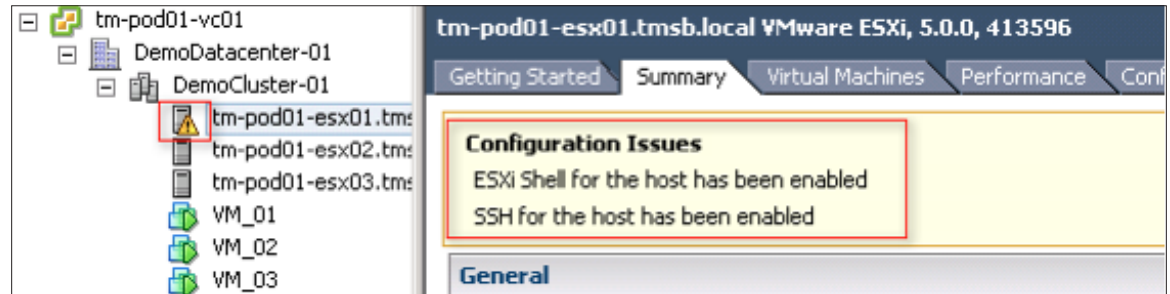


Figure 59. Notification That the ESXi Shell Has Been Enabled

Installing the vCLI

The vCLI is available on Microsoft Windows, Linux, and with the vMA virtual appliance.

Installing the vCLI on Windows

The vCLI installation package for Windows includes the ActivePerl runtime environment, along with the required Perl modules and libraries. The vCLI is supported on the following Windows platforms:

- Microsoft Windows Vista Enterprise SP1 32-bit and 64-bit
- Microsoft Windows 2008 64-bit
- Microsoft Windows 7 32-bit and 64-bit

To install the vCLI on Windows, download the vCLI installer package for Windows on a supported Windows server and launch the installer. Refer to Chapter 2 of the *Getting Started with vSphere Command-Line Interfaces* guide for information on how to install the vCLI on a Windows server.

Installing the vCLI on Linux

The vCLI installation package for Linux includes the vCLI scripts and the VMware vSphere 5.0 SDK for Perl. It can be installed on the Red Hat Enterprise Linux 5.5 server, SUSE Linux Enterprise 10 and 11 servers, and the Ubuntu 10.04 server. Download the vCLI package for your Linux distribution and run the installation script. Refer to Chapter 2 of the *Getting Started with vSphere Command-Line Interfaces* guide for information on how to install the vCLI on a Linux server.

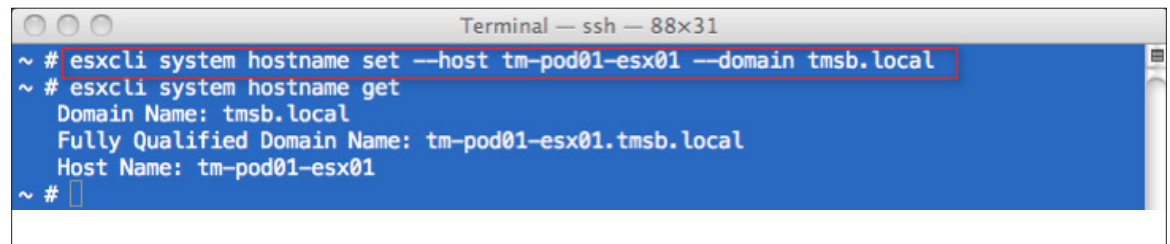
Installing the vCLI with the vMA

The vMA includes a Linux environment, the vCLI, and other prepackaged software. To install the vCLI with the vMA, simply deploy the vMA and log in to the console to configure the appliance. Refer to Chapter 2 of the *Getting Started with vSphere Command-Line Interfaces* guide for information on how to install and configure the vMA.

Sample esxcli Commands Run Locally from the ESXi Shell

The following examples show esxcli commands executed from the local ESXi Shell. Because they are being run from the ESXi Shell, it is not necessary to provide the server information or user credentials with the command.

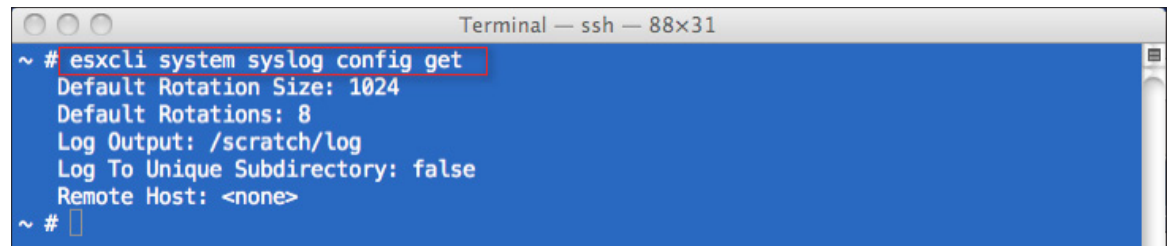
Use the **esxcli system hostname set --host tm-pod01-esx01 --domain tmsb.local** command to set the host name and domain name of the ESXi host. Then use the **esxcli system hostname get** command to display the host name and domain name and verify the change.

A terminal window titled "Terminal — ssh — 88x31" with a blue background. The prompt is "~ #". The first command is "esxcli system hostname set --host tm-pod01-esx01 --domain tmsb.local", which is highlighted with a red box. The second command is "esxcli system hostname get". The output shows "Domain Name: tmsb.local", "Fully Qualified Domain Name: tm-pod01-esx01.tmsb.local", and "Host Name: tm-pod01-esx01". The prompt returns to "~ #".

```
~ # esxcli system hostname set --host tm-pod01-esx01 --domain tmsb.local
~ # esxcli system hostname get
Domain Name: tmsb.local
Fully Qualified Domain Name: tm-pod01-esx01.tmsb.local
Host Name: tm-pod01-esx01
~ #
```

Figure 60. Set ESXi Host Name and Domain Name from the ESXi Shell

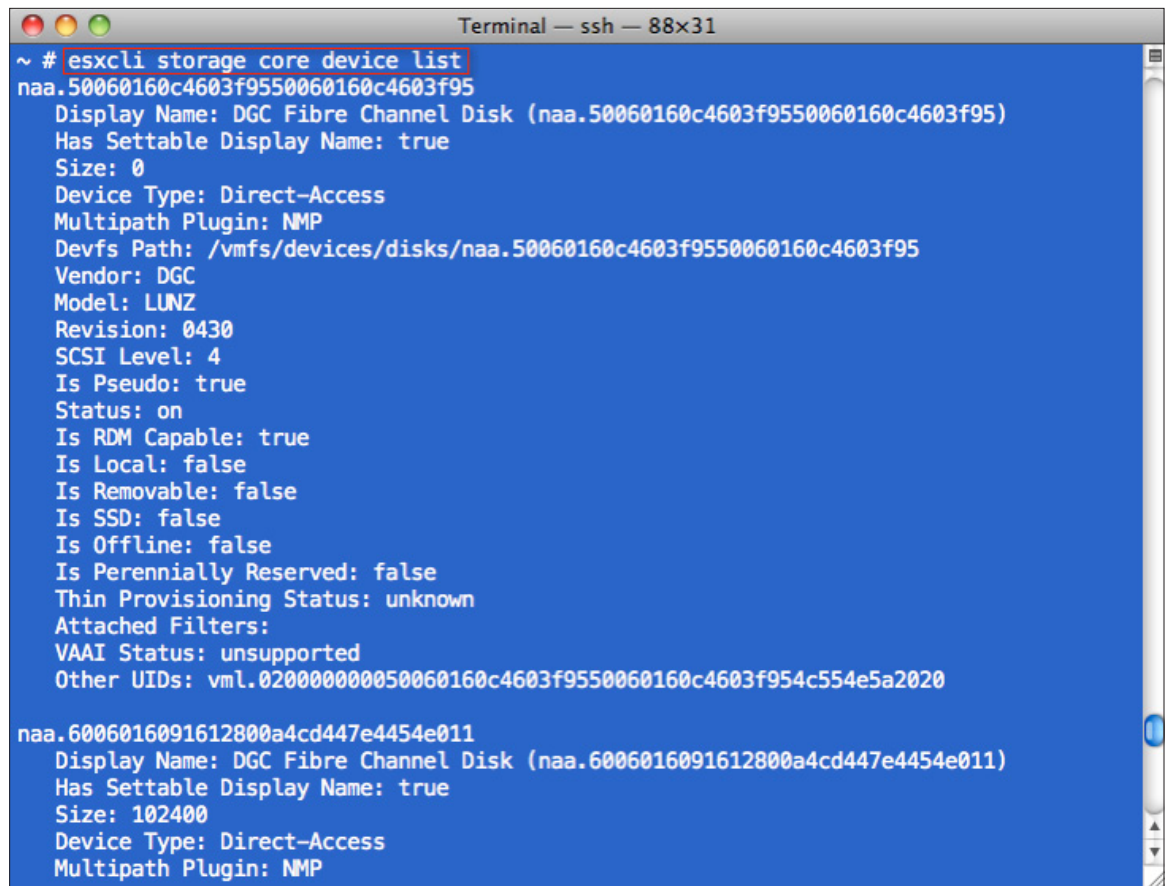
Use the **esxcli system syslog config get** command to display the ESXi host syslog configuration, as follows:

A terminal window titled "Terminal — ssh — 88x31" with a blue background. The prompt is "~ #". The command "esxcli system syslog config get" is highlighted with a red box. The output shows "Default Rotation Size: 1024", "Default Rotations: 8", "Log Output: /scratch/log", "Log To Unique Subdirectory: false", and "Remote Host: <none>". The prompt returns to "~ #".

```
~ # esxcli system syslog config get
Default Rotation Size: 1024
Default Rotations: 8
Log Output: /scratch/log
Log To Unique Subdirectory: false
Remote Host: <none>
~ #
```

Figure 61. Display Host Syslog Settings from the ESXi Shell

Use the **esxcli storage core device list** command to list all the storage devices on the ESXi host, as follows:



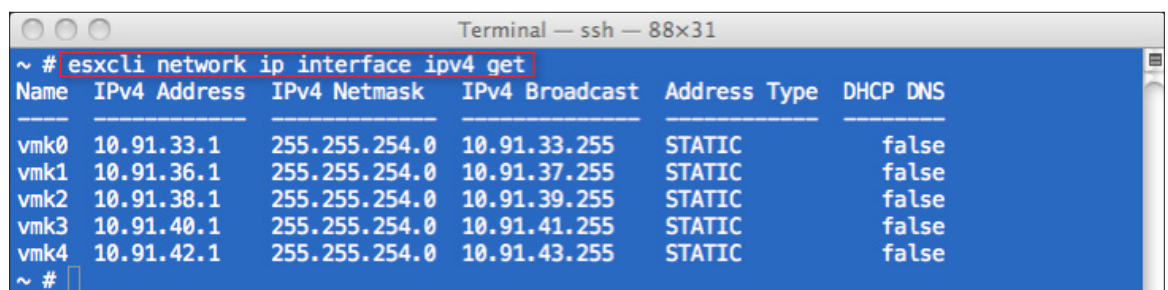
```

~ # esxcli storage core device list
naa.50060160c4603f9550060160c4603f95
  Display Name: DGC Fibre Channel Disk (naa.50060160c4603f9550060160c4603f95)
  Has Settable Display Name: true
  Size: 0
  Device Type: Direct-Access
  Multipath Plugin: NMP
  Devfs Path: /vmfs/devices/disks/naa.50060160c4603f9550060160c4603f95
  Vendor: DGC
  Model: LUNZ
  Revision: 0430
  SCSI Level: 4
  Is Pseudo: true
  Status: on
  Is RDM Capable: true
  Is Local: false
  Is Removable: false
  Is SSD: false
  Is Offline: false
  Is Perennially Reserved: false
  Thin Provisioning Status: unknown
  Attached Filters:
  VAAI Status: unsupported
  Other UUIDs: vml.02000000050060160c4603f9550060160c4603f954c554e5a2020

naa.6006016091612800a4cd447e4454e011
  Display Name: DGC Fibre Channel Disk (naa.6006016091612800a4cd447e4454e011)
  Has Settable Display Name: true
  Size: 102400
  Device Type: Direct-Access
  Multipath Plugin: NMP
  
```

Figure 62. Display Storage Devices from the ESXi Shell

Use the **esxcli network ip interface ipv4 get** command to list all the configured IPv4 addresses on the ESXi host, as follows:



```

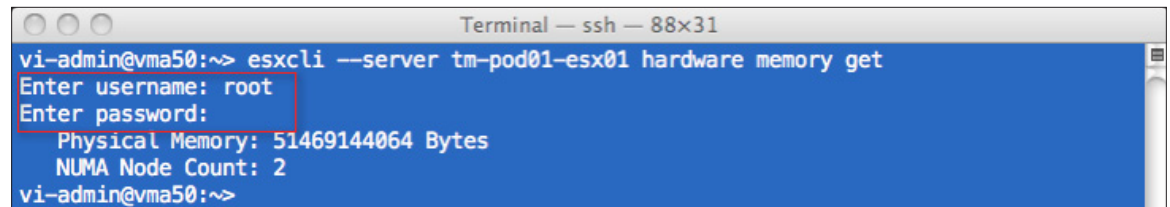
~ # esxcli network ip interface ipv4 get
Name      IPv4 Address  IPv4 Netmask  IPv4 Broadcast  Address Type  DHCP  DNS
-----
vmk0      10.91.33.1    255.255.254.0 10.91.33.255    STATIC        false
vmk1      10.91.36.1    255.255.254.0 10.91.37.255    STATIC        false
vmk2      10.91.38.1    255.255.254.0 10.91.39.255    STATIC        false
vmk3      10.91.40.1    255.255.254.0 10.91.41.255    STATIC        false
vmk4      10.91.42.1    255.255.254.0 10.91.43.255    STATIC        false
~ #
  
```

Figure 63. Display Configured IPs from the ESXi Shell

Sample esxcli Commands Run Remotely from the vCLI

The following examples show methods for using esxcli from the vCLI. For these examples, we will use the vMA. Because these commands are being run remotely, it is necessary to provide the **--server** and **--username** credentials as part of the esxcli command.

Use the **esxcli --server tm-pod01-esx01 hardware memory get** command to display the amount of memory on the ESXi hosts. Here we provide the **--server** option, but let it prompt for the user name and password, as follows:



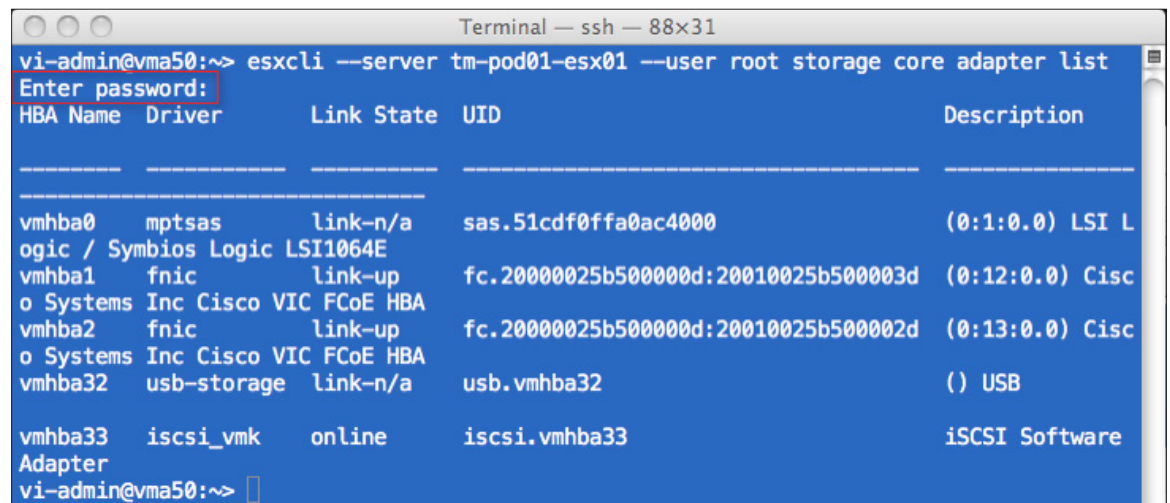
```

Terminal — ssh — 88x31
vi-admin@vma50:~> esxcli --server tm-pod01-esx01 hardware memory get
Enter username: root
Enter password:
Physical Memory: 51469144064 Bytes
NUMA Node Count: 2
vi-admin@vma50:~>

```

Figure 64. esxcli hardware memory get Command from vMA

Use the **esxcli --server tm-pod01-esx01 --user root storage core adapter list** command to list the available storage adapters on your host. Here we provide the **--server** and **--user** options, but let it prompt for the password, as follows:



```

Terminal — ssh — 88x31
vi-admin@vma50:~> esxcli --server tm-pod01-esx01 --user root storage core adapter list
Enter password:

```

HBA Name	Driver	Link State	UID	Description
vmhba0	mptsas	link-n/a	sas.51cdf0ffa0ac4000	(0:1:0.0) LSI L
ogic / Symbios Logic	LSI1064E			
vmhba1	fnic	link-up	fc.20000025b500000d:20010025b500003d	(0:12:0.0) Cisc
o Systems Inc	Cisco VIC FCoE HBA			
vmhba2	fnic	link-up	fc.20000025b500000d:20010025b500002d	(0:13:0.0) Cisc
o Systems Inc	Cisco VIC FCoE HBA			
vmhba32	usb-storage	link-n/a	usb.vmhba32	() USB
vmhba33	iscsi_vmk	online	iscsi.vmhba33	iSCSI Software

```

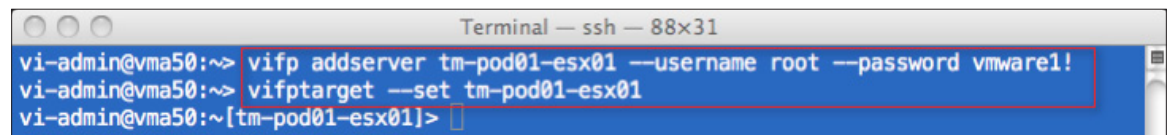
Adapter
vi-admin@vma50:~>

```

Figure 65. esxcli storage core adapter list Command from vMA

In this example, we use the vMA **-vi-fastpass** authentication, making it possible to run esxcli commands without providing the **-server**, **-username**, or **-password** options on the command line.

Start by setting up the vMA fastpass access, as follows:



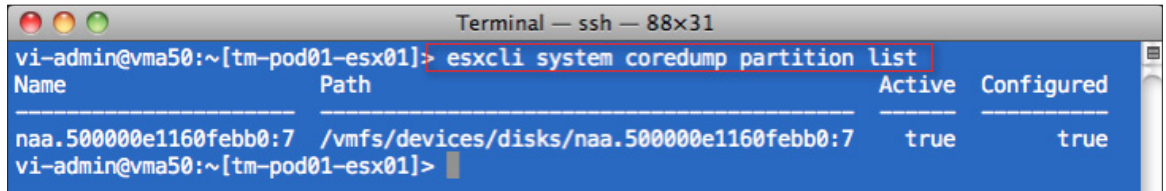
```

Terminal — ssh — 88x31
vi-admin@vma50:~> vifp addserver tm-pod01-esx01 --username root --password vmware1!
vi-admin@vma50:~> vifptarget --set tm-pod01-esx01
vi-admin@vma50:~[tm-pod01-esx01]>

```

Figure 66. Setting up vMA Fast Pass

With the fast pass target set to our ESXi host, we can now run the commands without specifying the options for the ESXi host, user name, or password. In the following example, we use the **esxcli system coredump partition list** command to show the configured core dump partition:



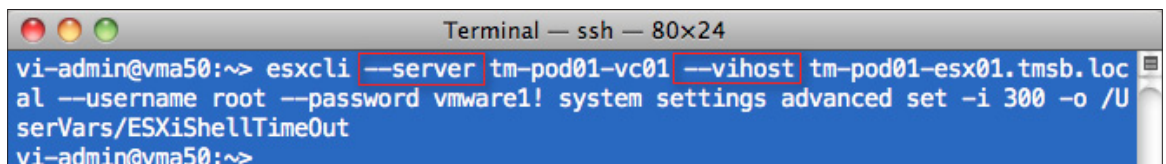
```

Terminal — ssh — 88x31
vi-admin@vma50:~[tm-pod01-esx01]> esxcli system coredump partition list
Name                                     Path                                     Active  Configured
-----
naa.500000e1160febb0:7                 /vmfs/devices/disks/naa.500000e1160febb0:7  true    true
vi-admin@vma50:~[tm-pod01-esx01]>

```

Figure 67. Display core dump partition list from vMA

In the following example, we will connect to the vCenter Server rather than connecting directly to the ESXi host. We will set the ESXi Shell timeout value to 300 seconds.



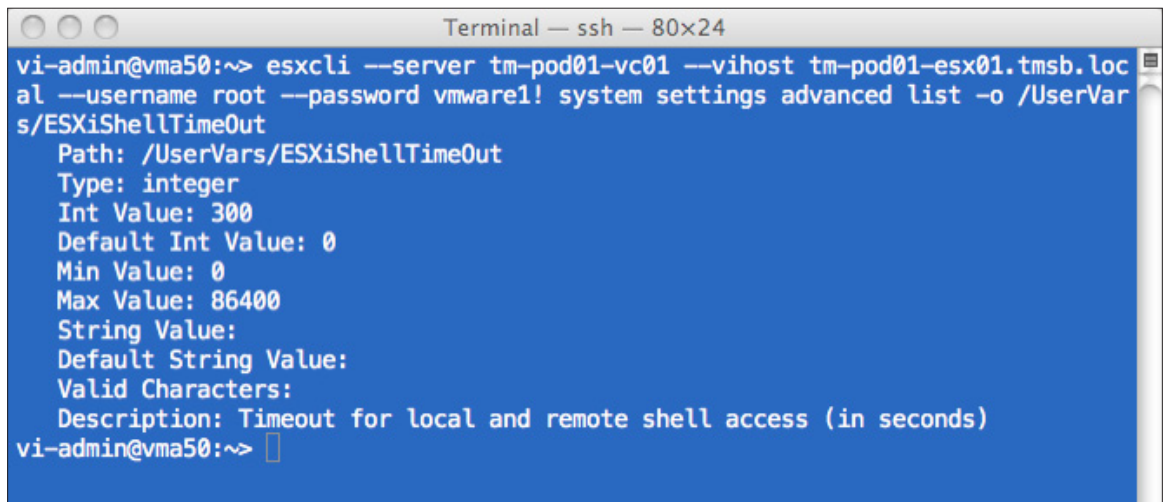
```

Terminal — ssh — 80x24
vi-admin@vma50:~> esxcli --server tm-pod01-vc01 --vihost tm-pod01-esx01.tmsb.local
--username root --password vmware1! system settings advanced set -i 300 -o /UserVars/ESXiShellTimeout
vi-admin@vma50:~>

```

Figure 68. Set ESXiShellTimeout

We can verify the change by displaying the new value of the ESXiShellTimeout, as follows:



```

Terminal — ssh — 80x24
vi-admin@vma50:~> esxcli --server tm-pod01-vc01 --vihost tm-pod01-esx01.tmsb.local
--username root --password vmware1! system settings advanced list -o /UserVars/ESXiShellTimeout
Path: /UserVars/ESXiShellTimeout
Type: integer
Int Value: 300
Default Int Value: 0
Min Value: 0
Max Value: 86400
String Value:
Default String Value:
Valid Characters:
Description: Timeout for local and remote shell access (in seconds)
vi-admin@vma50:~>

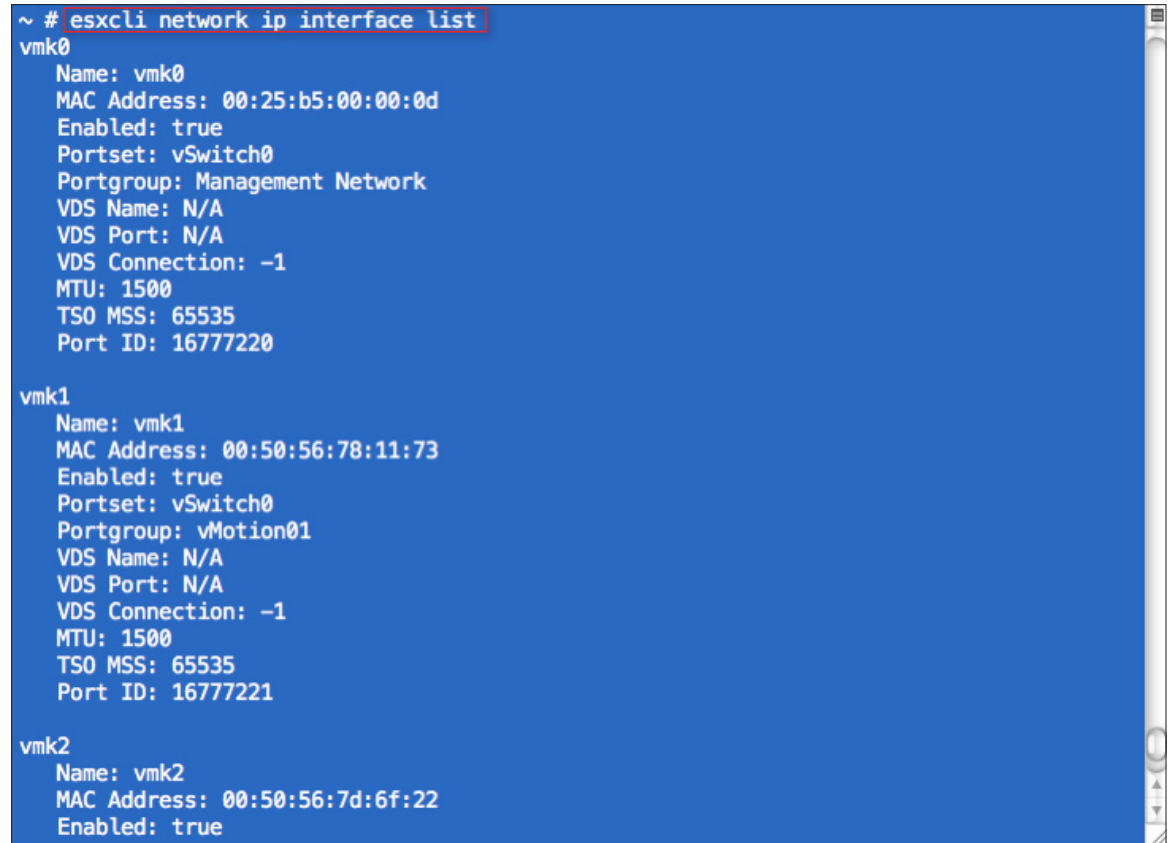
```

Figure 69. Display ESXiShellTimeout

Formatting esxcli Output

It is common to use the output of the esxcli command as input to another program or for inclusion in a report. To facilitate this, the esxcli command enables you to format and filter the command output in one of three formats: comma-separated values (CSV), key-value pair, or XML. In addition, you can specify which fields to include in the output.

In the following example, we need to generate a report showing all the configured interfaces on a host along with the vSwitch and port group to which they are assigned. We start by running the **esxcli network ip interface list** command, as follows:



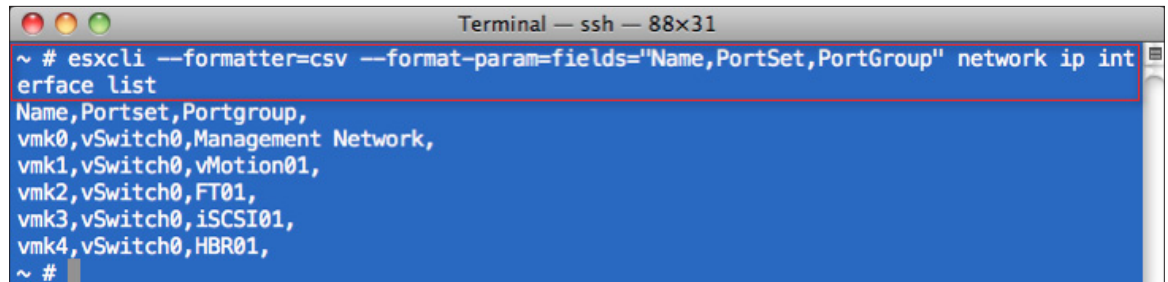
```
~ # esxcli network ip interface list
vmk0
  Name: vmk0
  MAC Address: 00:25:b5:00:00:0d
  Enabled: true
  Portset: vSwitch0
  Portgroup: Management Network
  VDS Name: N/A
  VDS Port: N/A
  VDS Connection: -1
  MTU: 1500
  TSO MSS: 65535
  Port ID: 16777220

vmk1
  Name: vmk1
  MAC Address: 00:50:56:78:11:73
  Enabled: true
  Portset: vSwitch0
  Portgroup: vMotion01
  VDS Name: N/A
  VDS Port: N/A
  VDS Connection: -1
  MTU: 1500
  TSO MSS: 65535
  Port ID: 16777221

vmk2
  Name: vmk2
  MAC Address: 00:50:56:7d:6f:22
  Enabled: true
```

Figure 70. esxcli network ip interface list Command from the ESXi Shell

The output gives us the information we need, but it is very verbose, requiring the user to use the scroll bar to see the data for all the interfaces. Because we need only a summary showing the interface name, vSwitch, and port group, we can refine our command using the **--formatter** and **--format-param** options, as follows:



```
Terminal — ssh — 88x31
~ # esxcli --formatter=csv --format-param=fields="Name,PortSet,PortGroup" network ip interface list
Name,Portset,Portgroup,
vmk0,vSwitch0,Management Network,
vmk1,vSwitch0,vMotion01,
vmk2,vSwitch0,FT01,
vmk3,vSwitch0,iSCSI01,
vmk4,vSwitch0,HBR01,
~ #
```

Figure 71. esxcli Command with -formatter Option from the ESXi Shell

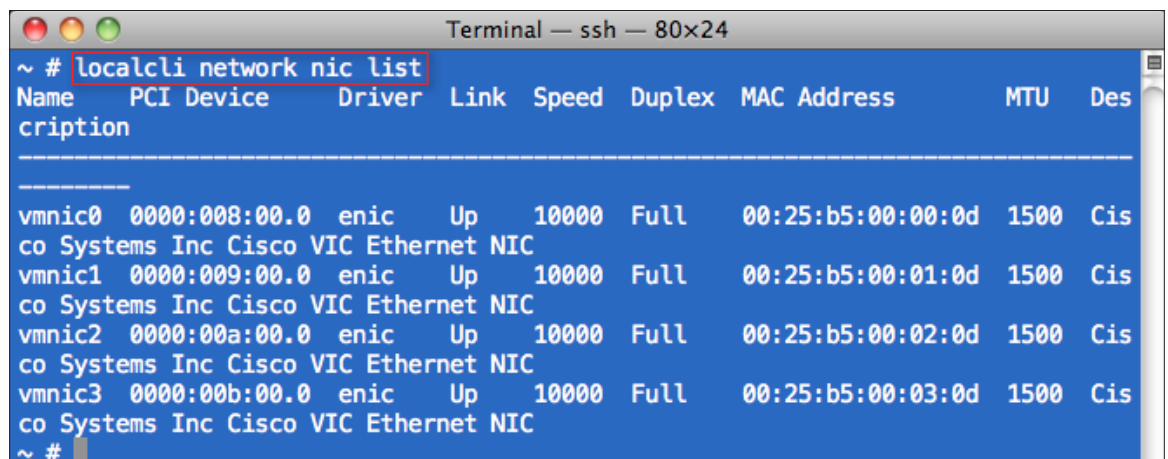
Now we have a short list giving us just the information we need.

The localcli Command

The **esxcli** command talks to the ESXi hosts through the **hostd** service. In rare circumstances, when the **hostd** service might not be responding, the **localcli** command can be used. The **localcli** command is equivalent to **esxcli** with the exception that it bypasses **hostd**. The **localcli** command is only intended for situations when **hostd** is unavailable and cannot be restarted. After you run the **localcli** command, you must restart **hostd**. Run **esxcli** commands after the restart.

*NOTE: Use the **localcli** command only under the direction of VMware technical support, because improper use can result in an inconsistent system state and potential failure of the ESXi host.*

The following example shows the use of the **localcli** command to display all network adapters on a host:



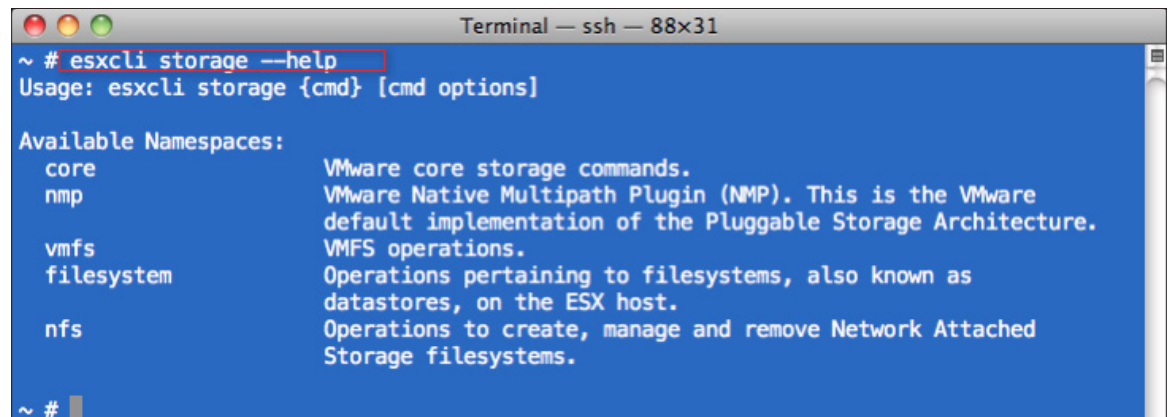
```
Terminal — ssh — 80x24
~ # localcli network nic list
Name      PCI Device      Driver  Link  Speed  Duplex  MAC Address      MTU  Description
-----
vmnic0 0000:008:00.0 enic    Up    10000  Full   00:25:b5:00:00:0d 1500 Cisco Systems Inc Cisco VIC Ethernet NIC
vmnic1 0000:009:00.0 enic    Up    10000  Full   00:25:b5:00:01:0d 1500 Cisco Systems Inc Cisco VIC Ethernet NIC
vmnic2 0000:00a:00.0 enic    Up    10000  Full   00:25:b5:00:02:0d 1500 Cisco Systems Inc Cisco VIC Ethernet NIC
vmnic3 0000:00b:00.0 enic    Up    10000  Full   00:25:b5:00:03:0d 1500 Cisco Systems Inc Cisco VIC Ethernet NIC
~ #
```

Figure 72. localcli Command Example

Bringing It All Together

The following example shows how to generate a list of the VMFS file systems on an ESXi host that have not been upgraded to VMFS-5. In this example, we will demonstrate the syntax discovery feature of `esxcli`.

Start by looking at the namespaces available under the storage namespace by running the `esxcli storage --help` command, as follows:

A terminal window titled "Terminal — ssh — 88x31" showing the command `~ # esxcli storage --help` and its output. The output lists available namespaces: core, nmp, vmfs, filesystem, and nfs, each with a brief description. The `filesystem` namespace is described as "Operations pertaining to filesystems, also known as datastores, on the ESX host." The `nfs` namespace is described as "Operations to create, manage and remove Network Attached Storage filesystems." The command prompt `~ #` is visible at the bottom.

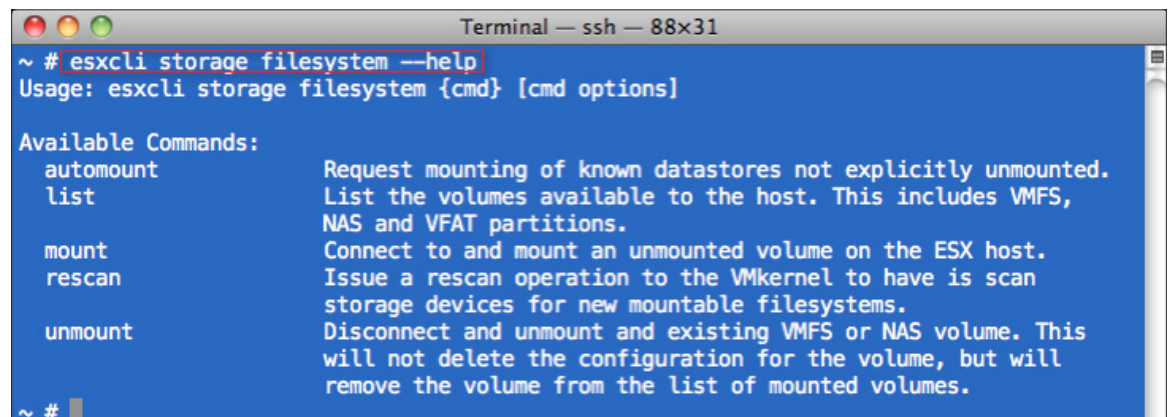
```
~ # esxcli storage --help
Usage: esxcli storage {cmd} [cmd options]

Available Namespaces:
  core          VMware core storage commands.
  nmp           VMware Native Multipath Plugin (NMP). This is the VMware
               default implementation of the Pluggable Storage Architecture.
  vmfs          VMFS operations.
  filesystem    Operations pertaining to filesystems, also known as
               datastores, on the ESX host.
  nfs          Operations to create, manage and remove Network Attached
               Storage filesystems.

~ #
```

Figure 73. `esxcli` Namespaces Under Storage from ESXi Shell

We see here that there is a `filesystem` name space. Next, we look to see what namespaces and commands are available under the `esxcli storage filesystem` namespace by running the `esxcli storage filesystem --help` command, as follows:

A terminal window titled "Terminal — ssh — 88x31" showing the command `~ # esxcli storage filesystem --help` and its output. The output lists available commands: automount, list, mount, rescan, and unmount, each with a brief description. The `list` command is described as "List the volumes available to the host. This includes VMFS, NAS and VFAT partitions." The `unmount` command is described as "Disconnect and unmount an existing VMFS or NAS volume. This will not delete the configuration for the volume, but will remove the volume from the list of mounted volumes." The command prompt `~ #` is visible at the bottom.

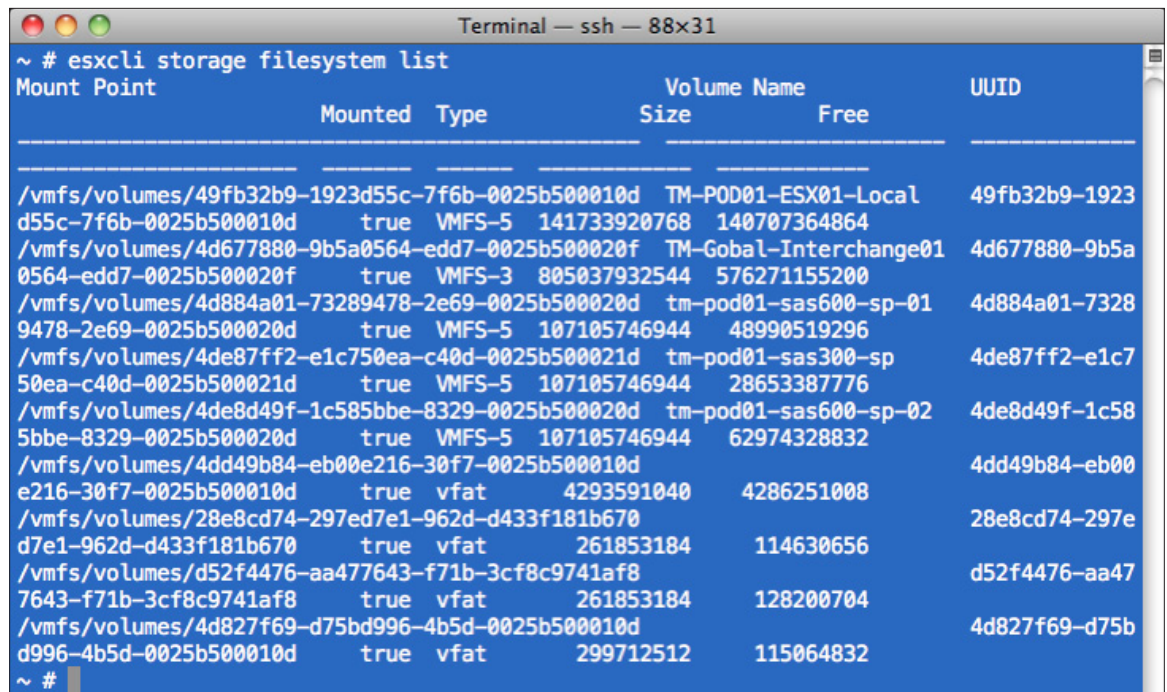
```
~ # esxcli storage filesystem --help
Usage: esxcli storage filesystem {cmd} [cmd options]

Available Commands:
  automount    Request mounting of known datastores not explicitly unmounted.
  list         List the volumes available to the host. This includes VMFS,
               NAS and VFAT partitions.
  mount        Connect to and mount an unmounted volume on the ESX host.
  rescan       Issue a rescan operation to the VMkernel to have it scan
               storage devices for new mountable filesystems.
  unmount      Disconnect and unmount an existing VMFS or NAS volume. This
               will not delete the configuration for the volume, but will
               remove the volume from the list of mounted volumes.

~ #
```

Figure 74. `esxcli` Namespaces Under Storage Filesystem from ESXi Shell

We see that there is a `list` command under the `filesystem` namespace that will list all the volumes on the host along with the VMFS information. We now run the final command, `esxcli storage filesystem list`, as follows:



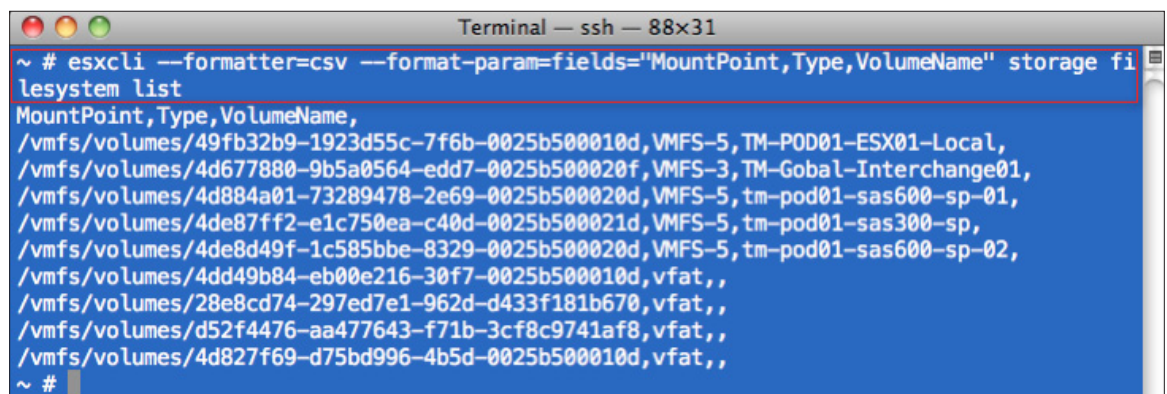
```

~ # esxcli storage filesystem list
Mount Point          Mounted  Type      Volume Name      Size      Free      UUID
-----
/vmfs/volumes/49fb32b9-1923d55c-7f6b-0025b500010d  true    VMFS-5    TM-P0D01-ESX01-Local  141733920768  140707364864  49fb32b9-1923d55c-7f6b-0025b500010d
/vmfs/volumes/4d677880-9b5a0564-edd7-0025b500020f  true    VMFS-3    TM-Gobal-Interchange01  805037932544  576271155200  4d677880-9b5a0564-edd7-0025b500020f
/vmfs/volumes/4d884a01-73289478-2e69-0025b500020d  true    VMFS-5    tm-pod01-sas600-sp-01  107105746944  48990519296  4d884a01-73289478-2e69-0025b500020d
/vmfs/volumes/4de87ff2-e1c750ea-c40d-0025b500021d  true    VMFS-5    tm-pod01-sas300-sp  107105746944  28653387776  4de87ff2-e1c750ea-c40d-0025b500021d
/vmfs/volumes/4de8d49f-1c585bbe-8329-0025b500020d  true    VMFS-5    tm-pod01-sas600-sp-02  107105746944  62974328832  4de8d49f-1c585bbe-8329-0025b500020d
/vmfs/volumes/4dd49b84-eb00e216-30f7-0025b500010d  true    vfat      4293591040  4286251008  4dd49b84-eb00e216-30f7-0025b500010d
/vmfs/volumes/28e8cd74-297ed7e1-962d-d433f181b670  true    vfat      261853184  114630656  28e8cd74-297ed7e1-962d-d433f181b670
/vmfs/volumes/d52f4476-aa477643-f71b-3cf8c9741af8  true    vfat      261853184  128200704  d52f4476-aa477643-f71b-3cf8c9741af8
/vmfs/volumes/4d827f69-d75bd996-4b5d-0025b500010d  true    vfat      299712512  115064832  4d827f69-d75bd996-4b5d-0025b500010d
~ #

```

Figure 75. esxcli storage filesystem list Command from ESXi Shell

This command gives us what we need. However, there is a lot of extra information in the output, making it hard to extrapolate the VMFS version information needed for our report. We can use the **--formatter** option with the **--format-param** filter to show only the information we need, as follows:



```

~ # esxcli --formatter=csv --format-param=fields="MountPoint,Type,VolumeName" storage filesystem list
MountPoint,Type,VolumeName,
/vmfs/volumes/49fb32b9-1923d55c-7f6b-0025b500010d,VMFS-5,TM-P0D01-ESX01-Local,
/vmfs/volumes/4d677880-9b5a0564-edd7-0025b500020f,VMFS-3,TM-Gobal-Interchange01,
/vmfs/volumes/4d884a01-73289478-2e69-0025b500020d,VMFS-5,tm-pod01-sas600-sp-01,
/vmfs/volumes/4de87ff2-e1c750ea-c40d-0025b500021d,VMFS-5,tm-pod01-sas300-sp,
/vmfs/volumes/4de8d49f-1c585bbe-8329-0025b500020d,VMFS-5,tm-pod01-sas600-sp-02,
/vmfs/volumes/4dd49b84-eb00e216-30f7-0025b500010d,vfat,,
/vmfs/volumes/28e8cd74-297ed7e1-962d-d433f181b670,vfat,,
/vmfs/volumes/d52f4476-aa477643-f71b-3cf8c9741af8,vfat,,
/vmfs/volumes/4d827f69-d75bd996-4b5d-0025b500010d,vfat,,
~ #

```

Figure 76. esxcli storage filesystem list with Formatting Command from ESXi Shell

We now have a list showing all the file systems on the ESXi host, along with the corresponding VMFS versions. From this we can easily identify those file systems that have not been upgraded to VMFS-5.

vSphere PowerCLI by Example

Introduction

vSphere PowerCLI is a snap-in (add-on) to Microsoft Windows PowerShell, a command-line scripting environment designed for Windows. It leverages the .NET object model, and was designed as an administrative language with system administrators in mind, because it provides administrators with easy-to-learn management and automation capabilities. vSphere PowerCLI adds over 200 cmdlets (commands) to native PowerShell commands, enabling the management of the vSphere environment.

Prerequisites

vSphere PowerCLI is typically installed on a vSphere administrator's Microsoft Windows-based desktop system. In order to support the installation of vSphere PowerCLI to a desktop system, the following prerequisite software packages must be present:

- Windows .NET Framework 3.5
- Windows PowerShell (V2 recommended)

Windows PowerShell V2 is integrated with Windows 7 and Windows 2008 R2. Previous operating systems, such as Windows XP, Windows Vista, Windows 2008 (not R2), and Windows 2003, are compatible with Windows PowerShell. This must be first downloaded and installed from the following Web site: <http://support.microsoft.com/kb/968929>

Install vSphere PowerCLI

After checking that all prerequisites are installed, you must set the execution policy of PowerShell to enable it to run scripts. By default, PowerShell is installed in secure mode, which will disable the running of scripts within PowerShell. To change the execution policy, start a PowerShell session with administrator privileges, as follows:

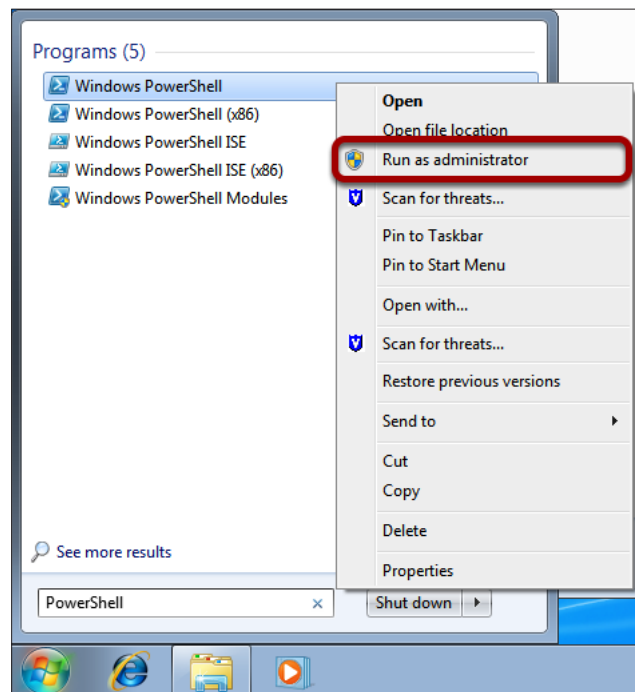


Figure 77. Starting PowerShell Session

From the Windows start menu, type **PowerShell**. Once the PowerShell program is displayed on the start menu, **right-click Windows PowerShell** and select **Run as administrator**.

A PowerShell prompt will be started, as follows:

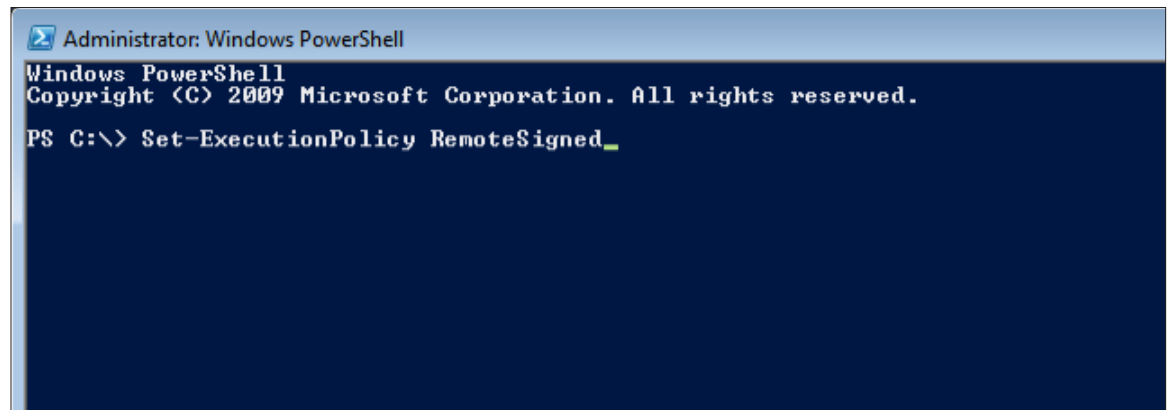


Figure 78. Setting Execution Policy

It is recommended that you read more about PowerShell's different execution policies to find out more information about these, and ensure that you change this to the correct setting for your organization. Enter **get-help about_Execution_Policies** at the PowerShell prompt.

In this guide, we will change the execution policy to RemoteSigned.

From the PowerShell prompt, enter **Set-ExecutionPolicy RemoteSigned**.

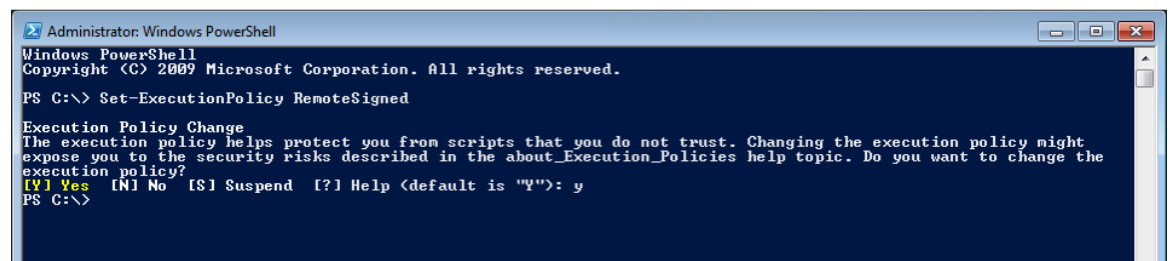


Figure 79. Information About Execution Policies

You will then receive information about execution policies and a prompt asking you to confirm your action before changing the execution policy. Enter **Y** at the prompt and **press Enter**.

You will then be returned to the PowerShell prompt with the change being completed. Type **Exit** and **press Enter** to leave the PowerShell prompt.

You are now ready to install vSphere PowerCLI.



Figure 80. vSphere PowerCLI Download Screen

Download the vSphere PowerCLI software to your workstation from the following URL: <http://vmware.com/go/PowerCLI>

Once the software has been downloaded, start the installation by **double-clicking the vSphere PowerCLI .exe file**.

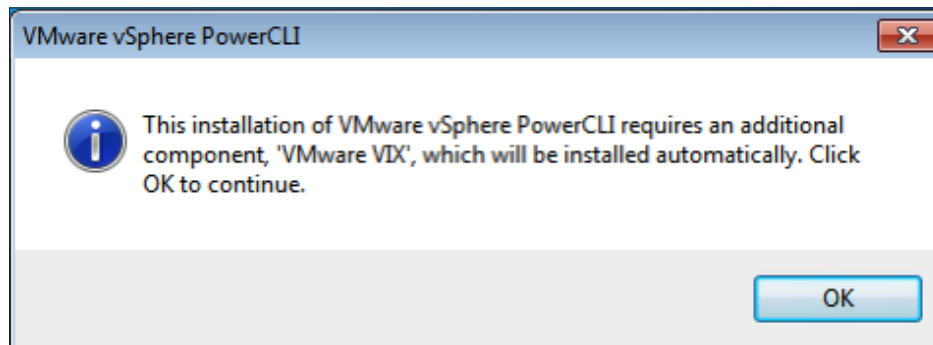


Figure 81. Notification of VMware VIX Installation

The installer will first notify you that an additional component, VMware VIX, will be installed as part of the vSphere PowerCLI installation. Click **OK**.

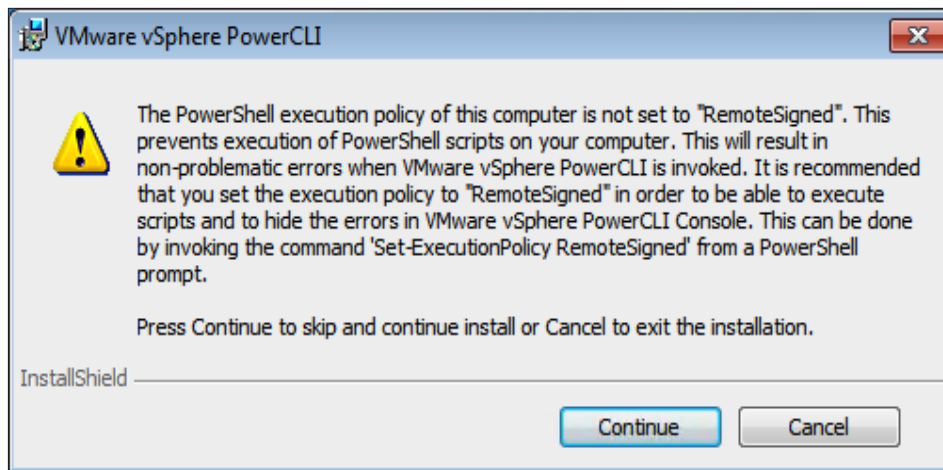


Figure 82. Recommendation to Set Execution Policy Correctly

If you have not yet set your execution policy correctly, an information box will appear advising you that this will need to be set to RemoteSigned before vSphere PowerCLI will execute correctly. Click the **Continue** button.

If the execution policy is set correctly this box will not appear.

This will bring you to the following **welcome screen**:



Figure 83. Welcome Screen

The welcome screen will now be shown, welcoming you to the install wizard for vSphere PowerCLI. Click **Next** to continue.

This will bring you to the following **VMware Patents screen**:

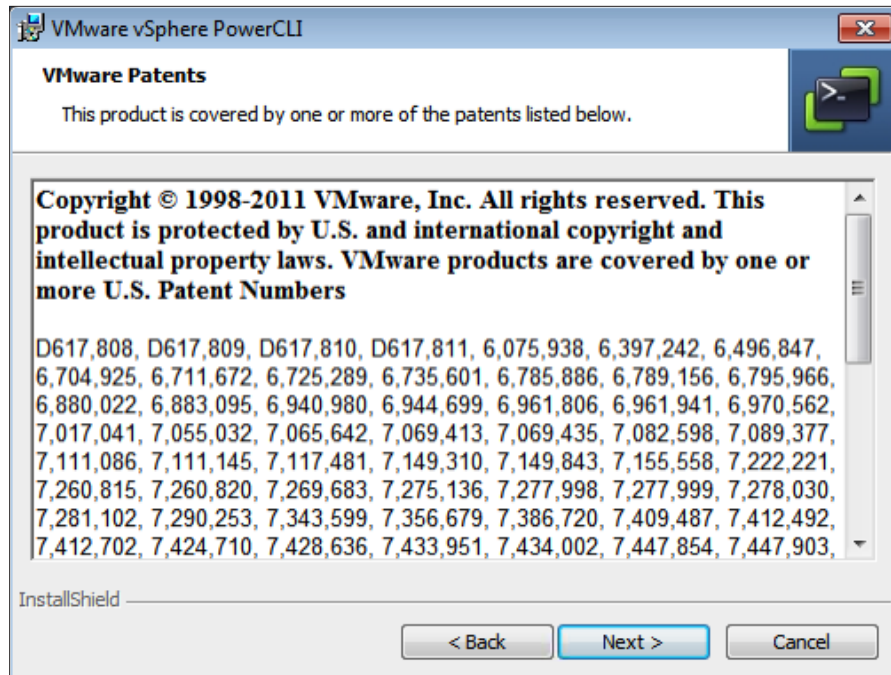


Figure 84. VMware Patents Screen

Click **Next** to continue. This will bring you to the following **License Agreement screen**:

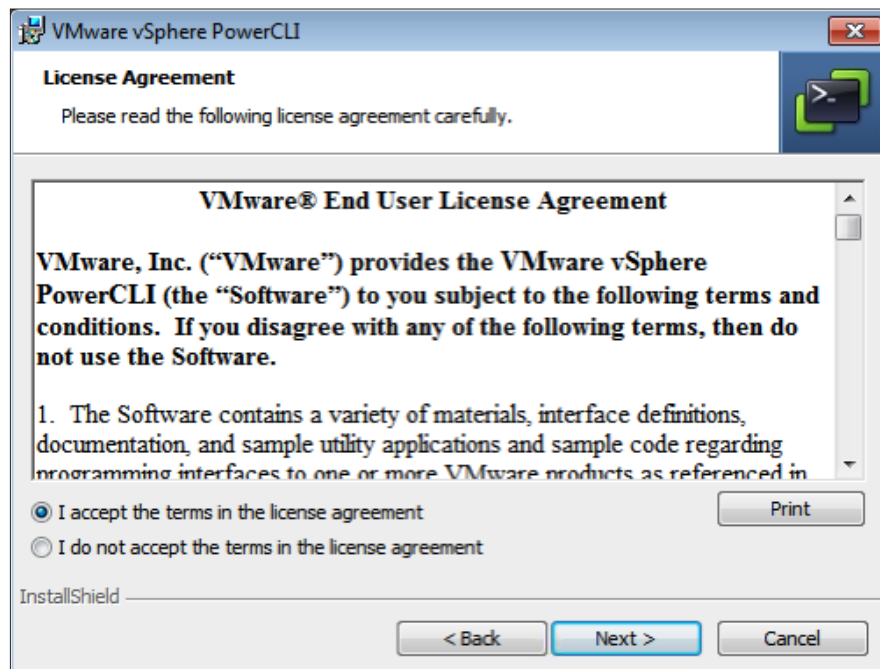


Figure 85. License Agreement Screen

Select the option, **I accept the terms in the license agreement** and then click **Next** to continue.

This will bring you to the following **Destination Folder screen**:

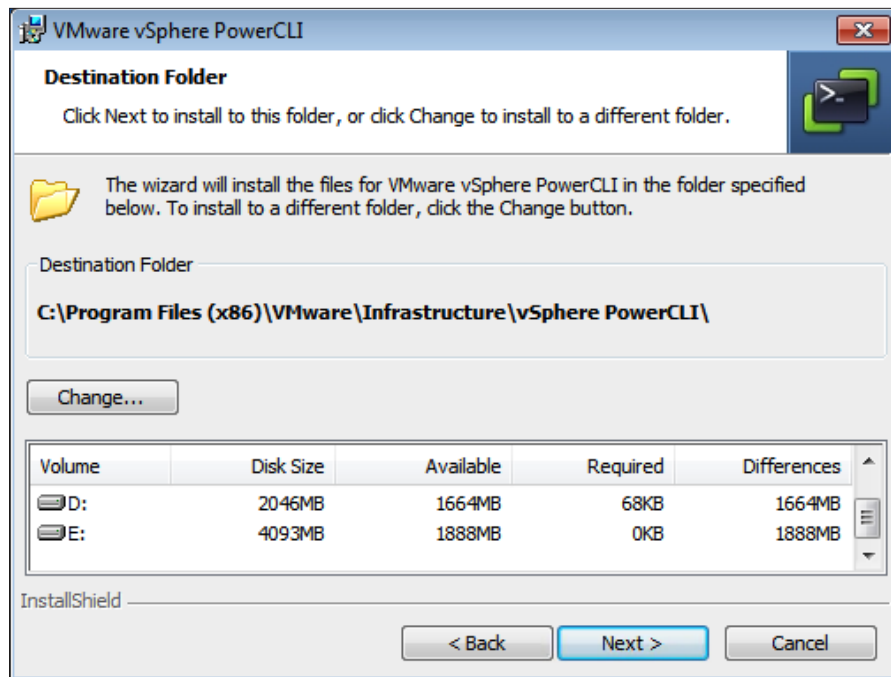


Figure 86. Destination Folder Screen

Select the drive you would like to install vSphere PowerCLI onto and the folder name, or leave this set as the recommended path and click **Next**.

This will bring you to the following **Ready to Install** screen:

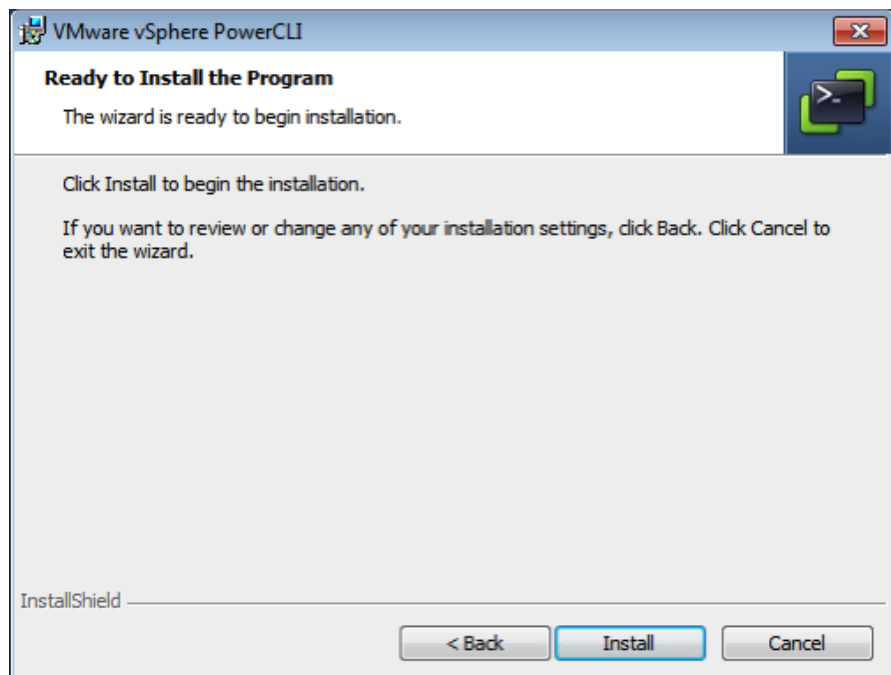


Figure 87. Ready to Install Screen

Click Install to begin the **Installation** of PowerCLI.

This will bring you to the **Installing VMware vSphere PowerCLI** screen.

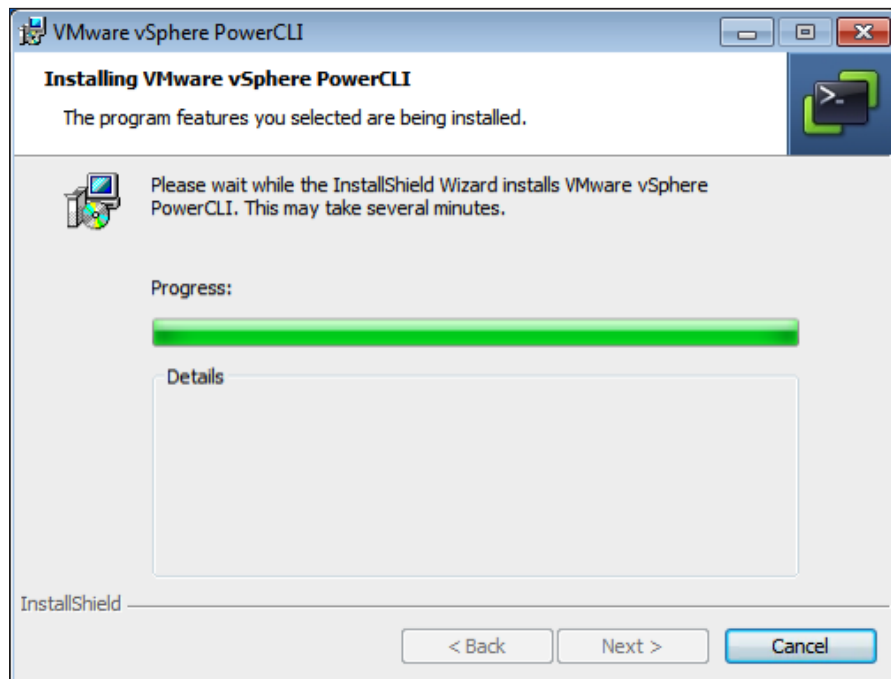


Figure 88. Installing vSphere PowerCLI

Wait while the installation is completed.

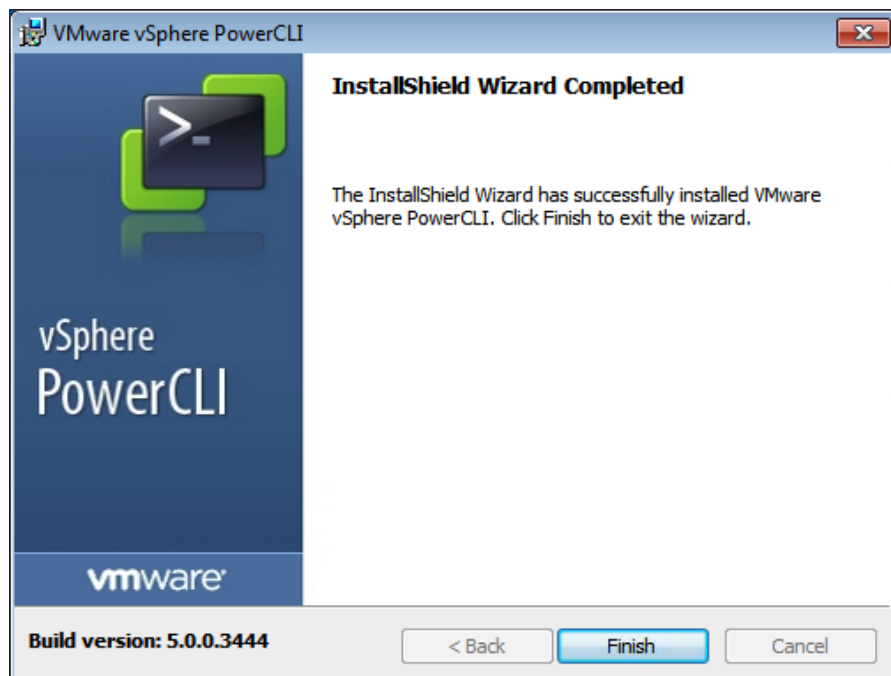


Figure 89. Installation Finish Screen

When the installation is successful, the **finish screen** will be displayed. To complete the installation, click **Finish**.

The vSphere PowerCLI installation will now be completed, and on your desktop you will now have two icons, which allow you to launch vSphere PowerCLI, a 64-bit version and a 32-bit version.

Getting Started with vSphere PowerCLI

On your start menu in the VMware -> VMware vSphere PowerCLI folder, you will now have access to the following items:

- vSphere PowerCLI (32-Bit)
- vSphere PowerCLI
- *vSphere PowerCLI Administration Guide*
- *vSphere PowerCLI Cmdlets Reference*
- *vSphere SDK for .NET API Reference*
- *vSphere SDK for .NET Dev Guide*

It is highly recommended that you read the *vSphere PowerCLI Administration Guide*, because this will provide the fundamentals of both vSphere PowerCLI and PowerShell, and will aid in the learning process when starting out with vSphere PowerCLI. This guide will show examples of vSphere PowerCLI and PowerShell code, but will not provide all knowledge to learn these languages in full. For further help and support, visit the vSphere PowerCLI community site at <http://vmware.com/go/PowerCLI>.

Connecting to a vSphere Host or vCenter

With vSphere PowerCLI, you have the ability, as with the vSphere Client, to connect to both vSphere hosts and vCenter servers. This document will show how to manage a vCenter Server and all connected entities, but it should be noted that the same cmdlets could be used to manage a single vSphere host.

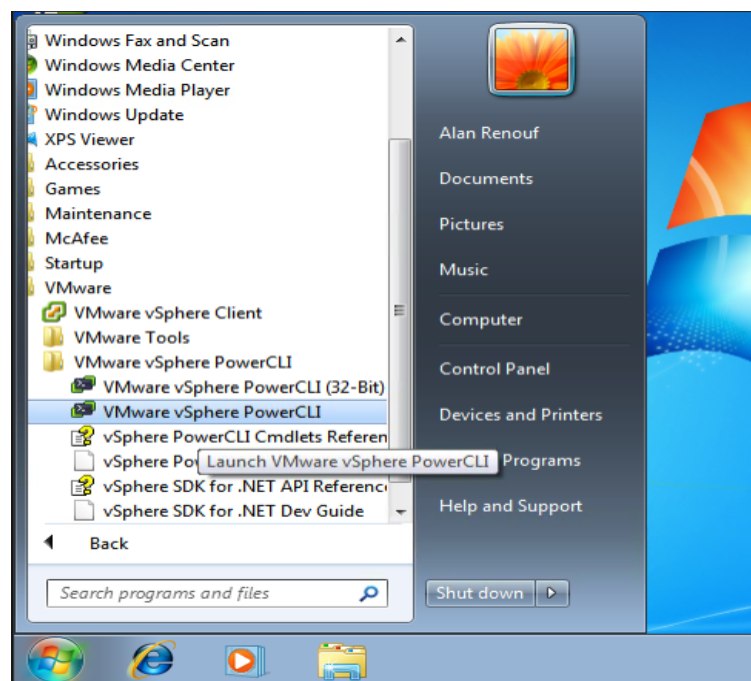


Figure 90. Launching PowerShell Session

From the start menu, select VMware -> VMware vSphere PowerCLI -> VMware vSphere PowerCLI.

This will launch a new PowerShell session and automatically import the VMware snap-in used to manage the VMware environment, as follows:

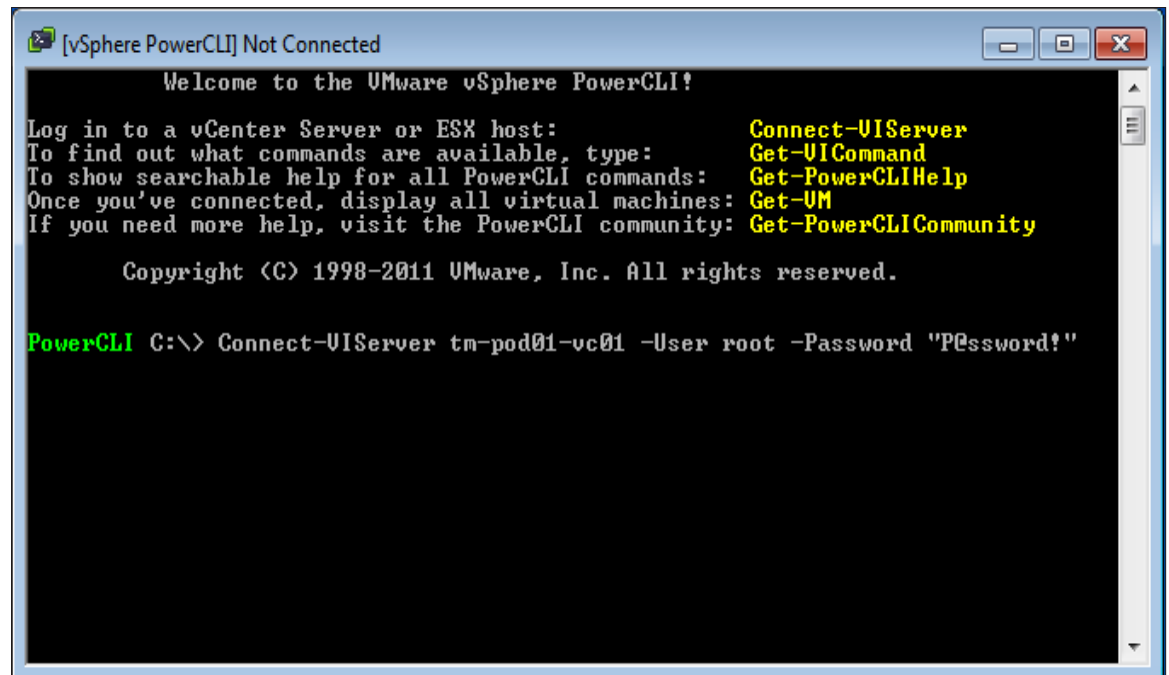
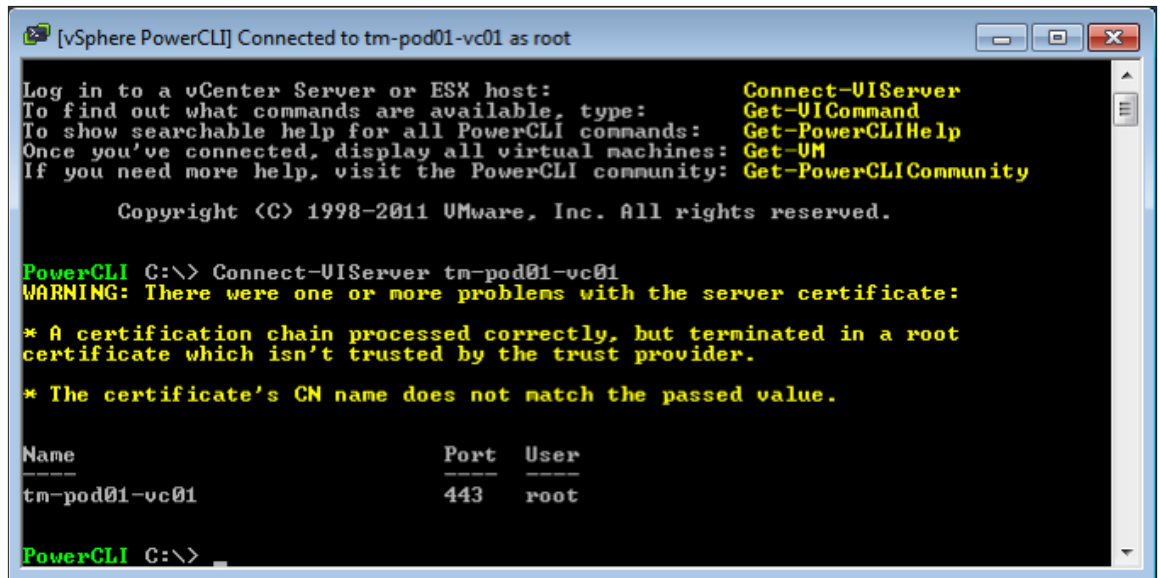


Figure 91. Connecting to vCenter Server

Use the **Connect-VIServer** cmdlet to connect to your vCenter Server. A user and password parameter can be used with this cmdlet to specify the connection credentials. If no user and password parameter is used, the cmdlet will try to log in with your current logged-on Windows credentials. If a connection cannot be made from the current credentials, you will be prompted for a user name and password.

Once connected, you will be returned to the vSphere PowerCLI prompt. You are then ready for your next cmdlet to be executed, as follows:



```
[vSphere PowerCLI] Connected to tm-pod01-vc01 as root

Log in to a vCenter Server or ESX host:          Connect-UIServer
To find out what commands are available, type:    Get-UICommand
To show searchable help for all PowerCLI commands: Get-PowerCLIHelp
Once you've connected, display all virtual machines: Get-VM
If you need more help, visit the PowerCLI community: Get-PowerCLICommunity

Copyright (C) 1998-2011 VMware, Inc. All rights reserved.

PowerCLI C:\> Connect-UIServer tm-pod01-vc01
WARNING: There were one or more problems with the server certificate:

* A certification chain processed correctly, but terminated in a root
certificate which isn't trusted by the trust provider.

* The certificate's CN name does not match the passed value.

Name                                Port    User
----                                -
tm-pod01-vc01                       443     root

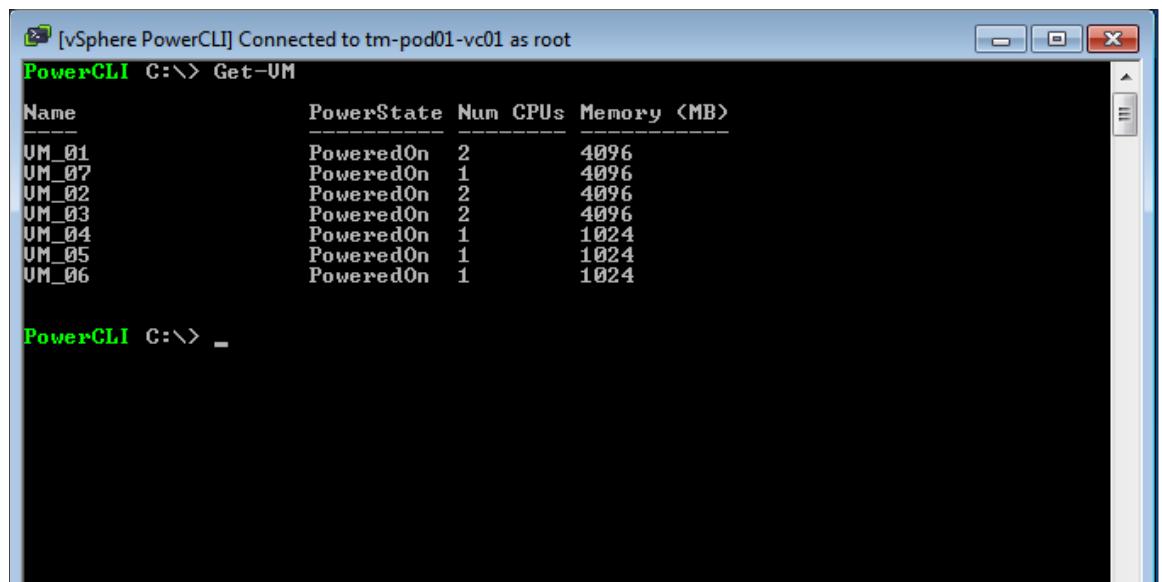
PowerCLI C:\> _
```

Figure 92. Certificate Warning

During this “vSphere PowerCLI by Example” section, the certificate warning can be ignored.

Once you are connected, the Name, Port and User properties used to make the connection will be returned to show a successful connection.

Using vSphere PowerCLI



```
[vSphere PowerCLI] Connected to tm-pod01-vc01 as root

PowerCLI C:\> Get-VM

Name                                PowerState Num CPUs Memory (MB)
----                                -
VM_01                               PoweredOn  2      4096
VM_07                               PoweredOn  1      4096
VM_02                               PoweredOn  2      4096
VM_03                               PoweredOn  2      4096
VM_04                               PoweredOn  1      1024
VM_05                               PoweredOn  1      1024
VM_06                               PoweredOn  1      1024

PowerCLI C:\> _
```

Figure 93.

To retrieve a list of virtual machines attached to the connected vCenter server, type **Get-VM**. This will return the Name, PowerState, Num CPUs and Memory (MB). These are all called properties of the virtual machine. vSphere PowerCLI returns more information than what is shown on the screen. It actually returns an object to this vSphere PowerCLI session containing more information about the virtual machine.

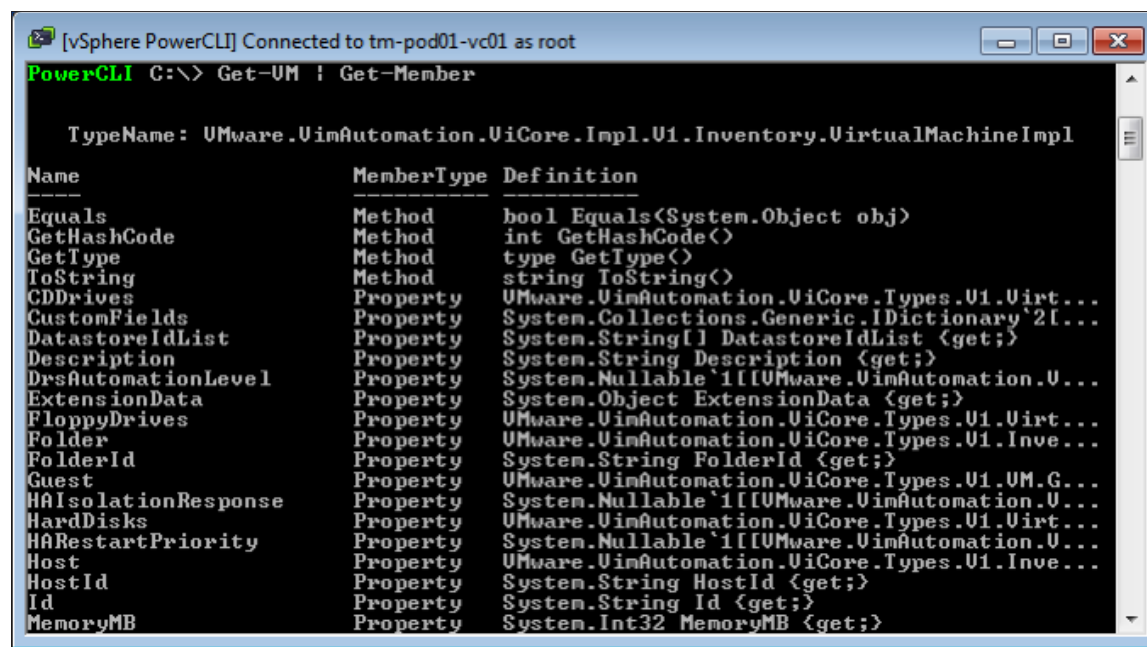


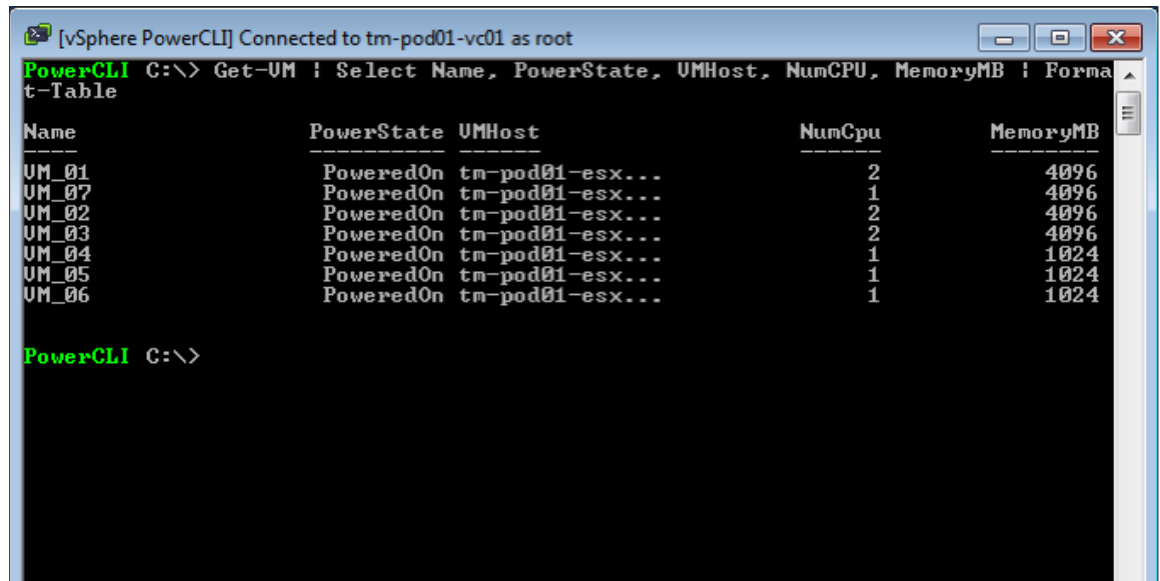
Figure 94.

To find out more information about the object being returned by vSphere PowerCLI, use the **Get-Member** cmdlet to retrieve a list of all properties, and also methods attached to this virtual machine object.

To do this we will take the **Get-VM** cmdlet and “pipe” it through the **Get-Member** cmdlet. This will take the results of the **Get-VM** cmdlet and push them as an input into the **Get-Member** cmdlet.

Type **Get-VM | Get-Member**

As you can see from the preceding screenshot, the virtual machine object contains more properties than were shown from our initial **Get-VM** results.



```
[vSphere PowerCLI] Connected to tm-pod01-vc01 as root
PowerCLI C:\> Get-VM | Select Name, PowerState, VMHost, NumCPU, MemoryMB | Format-Table
```

Name	PowerState	VMHost	NumCpu	MemoryMB
VM_01	PoweredOn	tm-pod01-esx...	2	4096
VM_07	PoweredOn	tm-pod01-esx...	1	4096
VM_02	PoweredOn	tm-pod01-esx...	2	4096
VM_03	PoweredOn	tm-pod01-esx...	2	4096
VM_04	PoweredOn	tm-pod01-esx...	1	1024
VM_05	PoweredOn	tm-pod01-esx...	1	1024
VM_06	PoweredOn	tm-pod01-esx...	1	1024

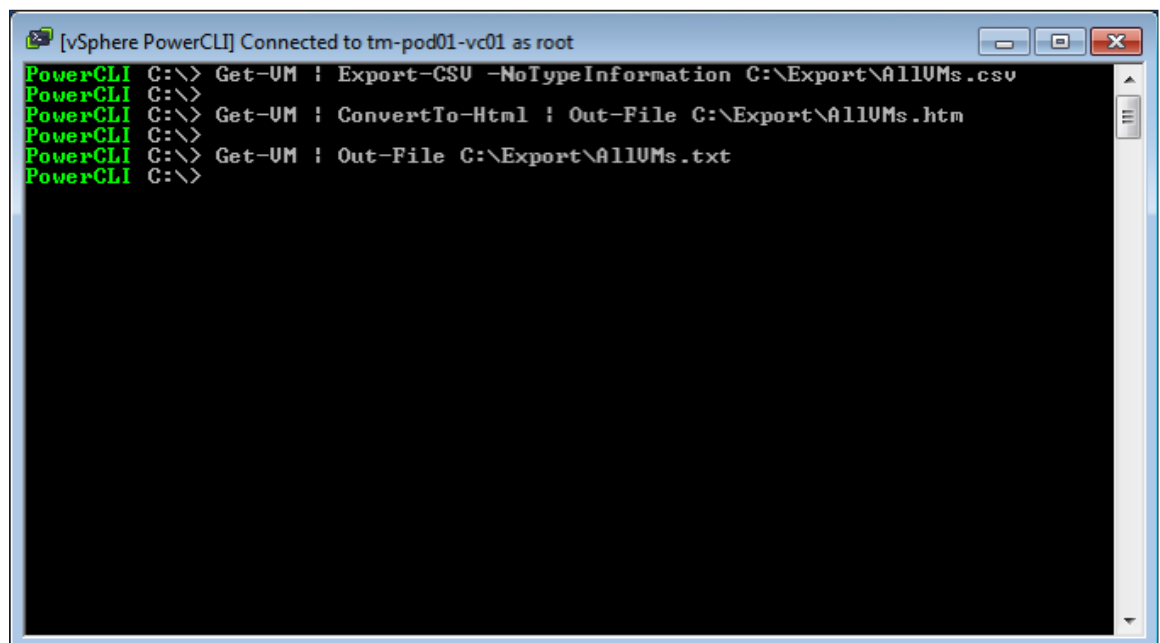
```
PowerCLI C:\>
```

Figure 95.

To select properties that we would like to see, we can use the **Select-Object** cmdlet to choose the properties of the virtual machine we would like returned.

Type: **Get-VM | Select Name, PowerState, VMHost, NumCPU, MemoryMB | Format-Table**

This will retrieve the selected properties and show them in a table view in our console.



```
[vSphere PowerCLI] Connected to tm-pod01-vc01 as root
PowerCLI C:\> Get-VM | Export-CSV -NoTypeInfo C:\Export\AllVMs.csv
PowerCLI C:\>
PowerCLI C:\> Get-VM | ConvertTo-Html | Out-File C:\Export\AllVMs.htm
PowerCLI C:\>
PowerCLI C:\> Get-VM | Out-File C:\Export\AllVMs.txt
PowerCLI C:\>
```

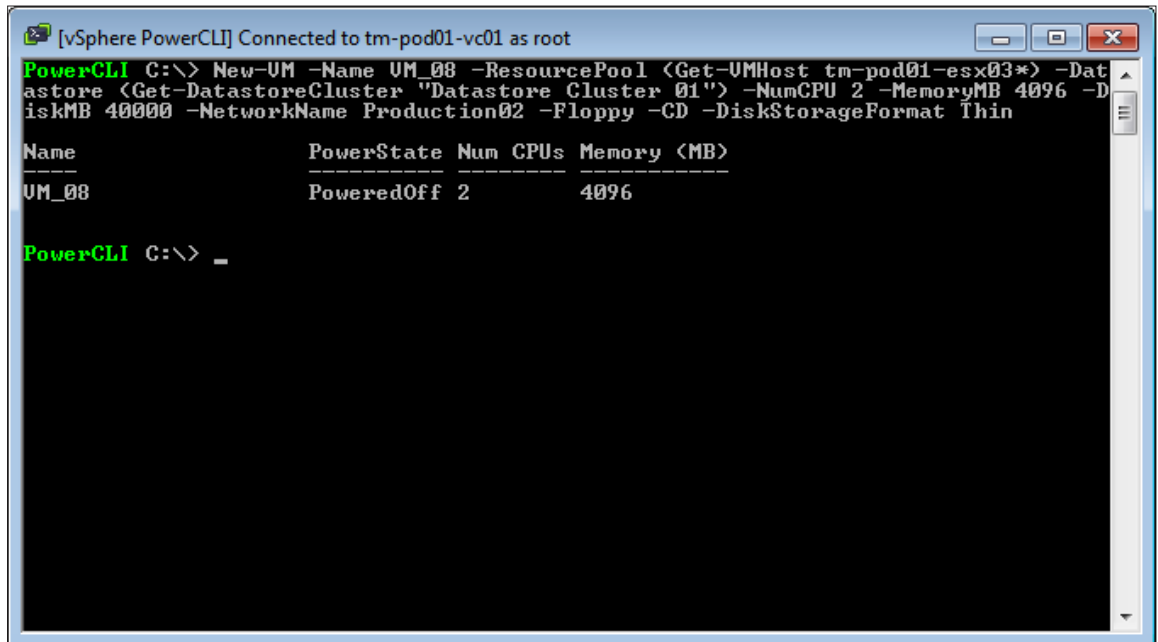
Figure 96.

This information can easily be exported from vSphere PowerCLI into many formats using some of the cmdlets built into the default PowerShell console.

To export the information into a comma-separated values file, type **Get-VM | Export-CSV -NoTypeInfoInformation C:\Export\AllVMs.csv**

To export the information into a html file, type **Get-VM | ConvertTo-Html | Out-File C:\Export\AllVMs.htm**

To export the information into a plain text file, type **Get-VM | Out-File C:\Export\AllVMs.txt**



```
[vSphere PowerCLI] Connected to tm-pod01-vc01 as root
PowerCLI C:\> New-VM -Name VM_08 -ResourcePool <Get-VMHost tm-pod01-esx03*> -Datastore <Get-DatastoreCluster "Datastore Cluster 01"> -NumCPU 2 -MemoryMB 4096 -DiskMB 40000 -NetworkName Production02 -Floppy -CD -DiskStorageFormat Thin
```

Name	PowerState	Num CPUs	Memory (MB)
VM_08	PoweredOff	2	4096

```
PowerCLI C:\> _
```

Figure 97.

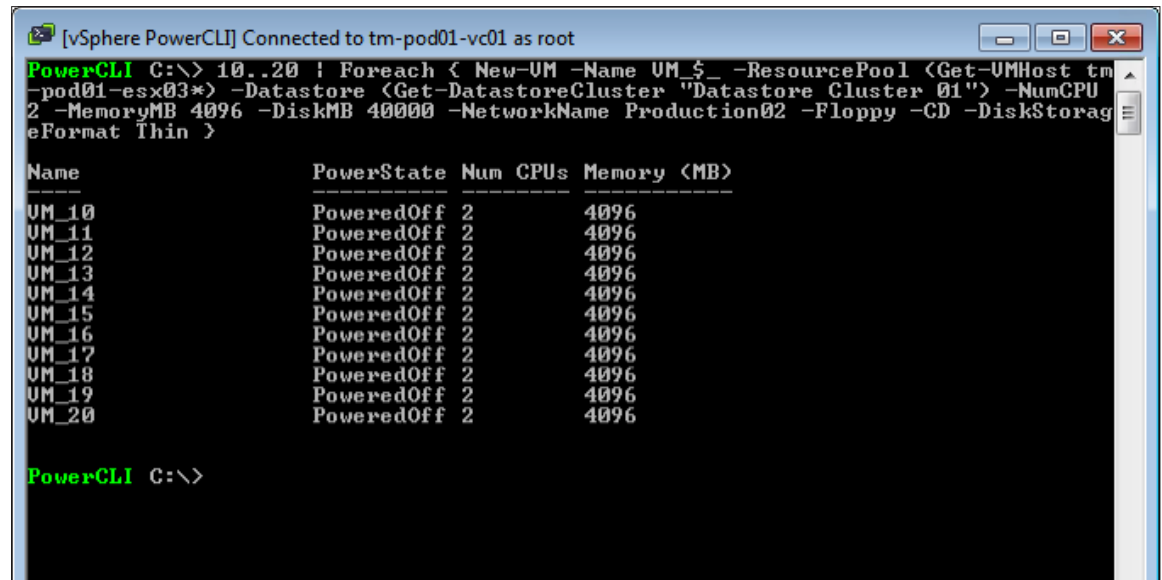
To create a new virtual machine, the **New-VM** cmdlet can be used. This has many parameters that can be used to specify the exact configuration of the virtual machine. To view these parameters, use the **Get-Help** cmdlet.

We will create a virtual machine with the following configuration:

- Name: VM_08
- Host: tm-pod01-esx03
- Datastore: Datastore Cluster 01
- CPUs: 2
- Memory: 4GB
- Disk: 40GB
- DiskType: Thin
- Network: Production02
- Floppy Drive: Yes
- CD-Rom: Yes

To do this, type the following:

New-VM -Name VM_08 -ResourcePool (Get-VMHost tm-pod01-esx03*) -Datastore (Get-DatastoreCluster "Datastore Cluster 01") -NumCPU 2 -MemoryMB 4096 -DiskMB 40000 -NetworkName Production02 -Floppy -CD -DiskStorageFormat Thin



```
[vSphere PowerCLI] Connected to tm-pod01-vc01 as root
PowerCLI C:\> 10..20 | Foreach { New-VM -Name VM_$ -ResourcePool (Get-VMHost tm-pod01-esx03*) -Datastore (Get-DatastoreCluster "Datastore Cluster 01") -NumCPU 2 -MemoryMB 4096 -DiskMB 40000 -NetworkName Production02 -Floppy -CD -DiskStorageFormat Thin }
```

Name	PowerState	Num CPUs	Memory (MB)
VM_10	PoweredOff	2	4096
VM_11	PoweredOff	2	4096
VM_12	PoweredOff	2	4096
VM_13	PoweredOff	2	4096
VM_14	PoweredOff	2	4096
VM_15	PoweredOff	2	4096
VM_16	PoweredOff	2	4096
VM_17	PoweredOff	2	4096
VM_18	PoweredOff	2	4096
VM_19	PoweredOff	2	4096
VM_20	PoweredOff	2	4096

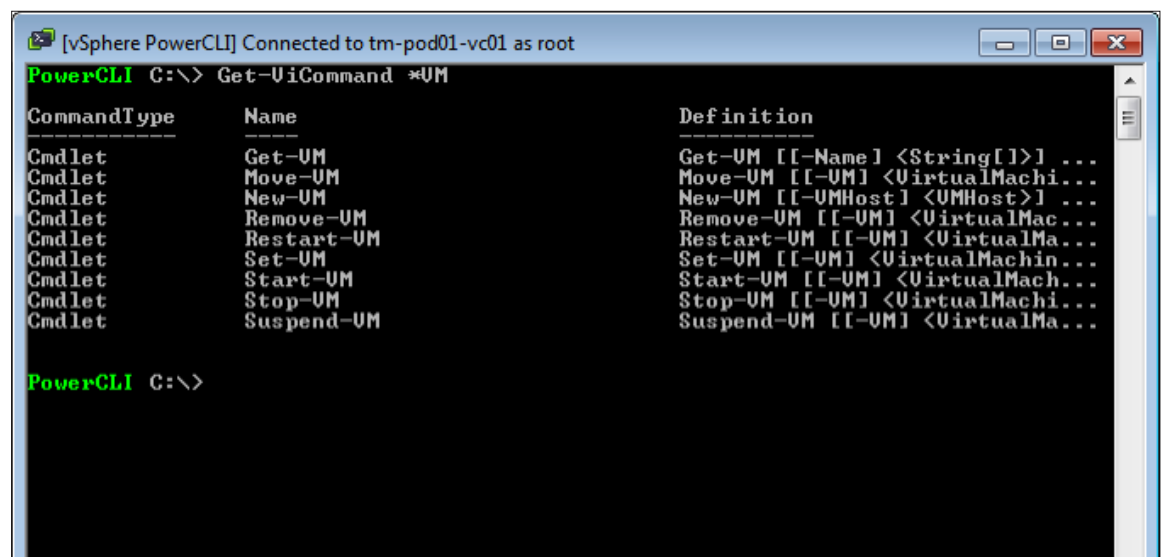
```
PowerCLI C:\>
```

Figure 98.

The New-VM cmdlet can also be used to create any number of virtual machines with the same configuration.

The following example shows how to create 10 new virtual machines with the same configuration. In the following example, \$ _ refers to the current number in the pipeline, because they are passed through to the New-VM cmdlet:

10..20 | Foreach { New-VM -Name VM_\$ -ResourcePool (Get-VMHost tm-pod01-esx03*) -Datastore (Get-DatastoreCluster "Datastore Cluster 01") -NumCPU 2 -MemoryMB 4096 -DiskMB 40000 -NetworkName Production02 -Floppy -CD -DiskStorageFormat Thin }



```
[vSphere PowerCLI] Connected to tm-pod01-vc01 as root
PowerCLI C:\> Get-UiCommand *VM
```

CommandType	Name	Definition
Cmdlet	Get-VM	Get-VM [[-Name] <String[]>] ...
Cmdlet	Move-VM	Move-VM [[-VM] <VirtualMachi...]
Cmdlet	New-VM	New-VM [[-VMHost] <VMHost>] ...
Cmdlet	Remove-VM	Remove-VM [[-VM] <VirtualMac...]
Cmdlet	Restart-VM	Restart-VM [[-VM] <VirtualMa...]
Cmdlet	Set-VM	Set-VM [[-VM] <VirtualMachin...]
Cmdlet	Start-VM	Start-VM [[-VM] <VirtualMach...]
Cmdlet	Stop-VM	Stop-VM [[-VM] <VirtualMachi...]
Cmdlet	Suspend-VM	Suspend-VM [[-VM] <VirtualMa...]

```
PowerCLI C:\>
```

Figure 99.

Further virtual machine operations can be performed with vSphere PowerCLI. To see the cmdlets that can be used with virtual machines, type **Get-ViCommand *VM**

To find more information on one of these cmdlets, type **Get-Help Move-VM -Full**

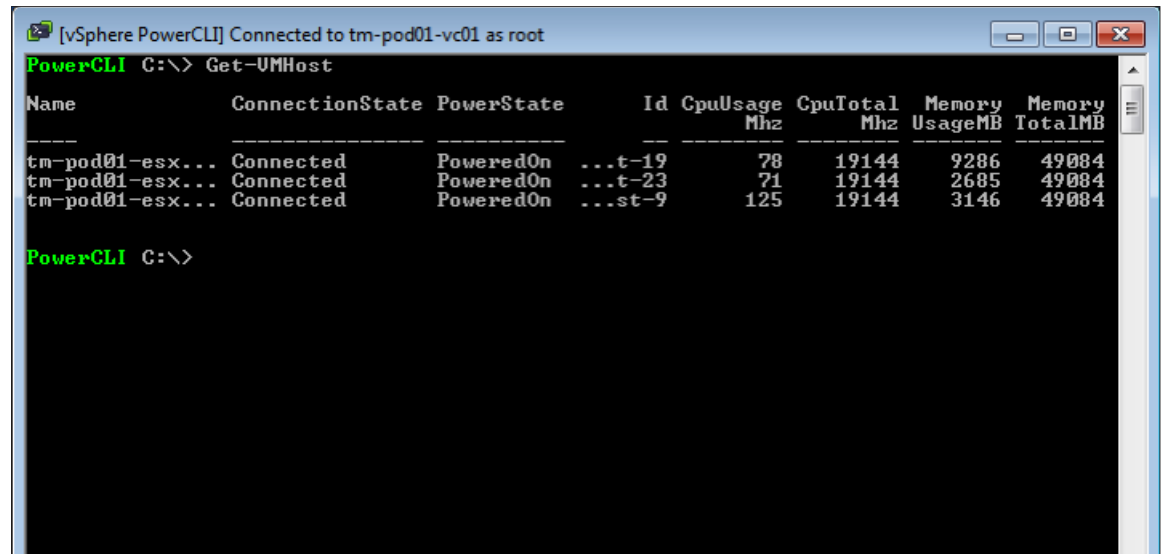


Figure 100.

To list all hosts attached to the current connection, type **Get-VMHost**

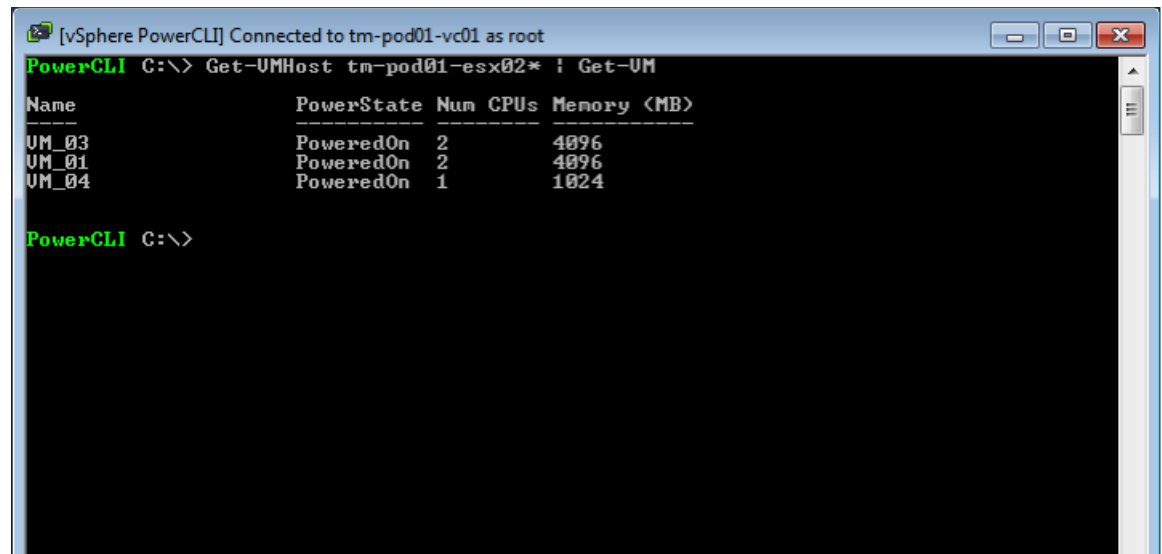


Figure 101.

To list all virtual machines attached to a certain host, type **Get-VMHost tm-pod01-esx02* | Get-VM**

The preceding example will take the result of the **Get-VMHost** cmdlet and push it through as an input for the **Get-VM** cmdlet, producing a list of virtual machines on that specific host.

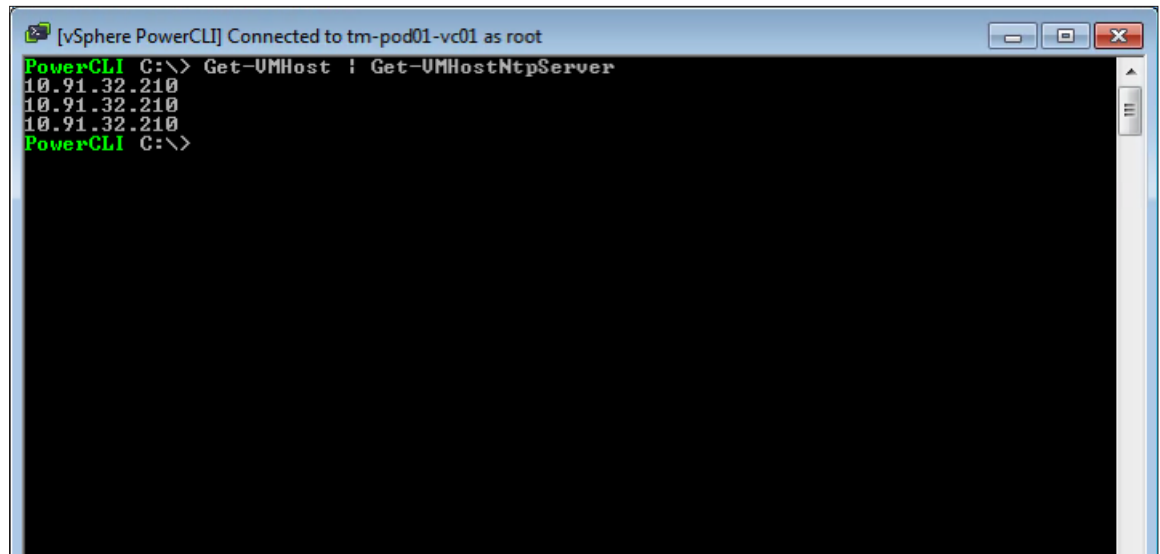


Figure 102.

The **Get-VMHost** cmdlet can be used in conjunction with other cmdlets to retrieve and set information for that host.

To list the NTP servers on each host in the vSphere Client, you would need to go to the host and clusters view, select a host, click the configuration tab and select the time configuration setting to view. You would need to repeat this for each host.

To do this in vSphere PowerCLI, type **Get-VMHost | Get-VMHostNTPServer**

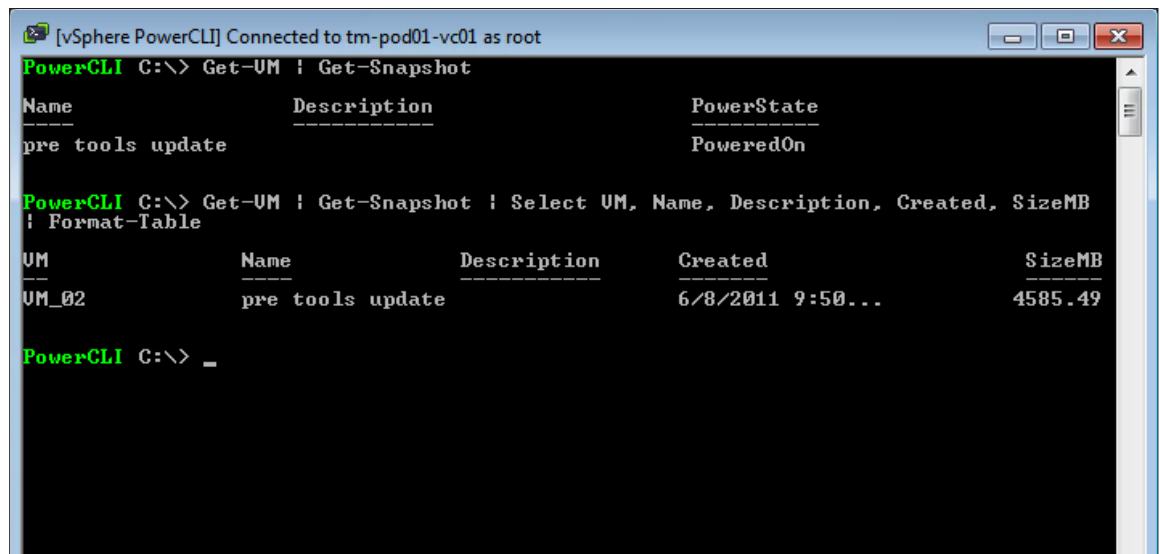


Figure 103.

Snapshot information gathering is a time-consuming part of any virtual administrator's job. Snapshots need to be managed correctly or they can quickly cause issues within the virtual infrastructure. Within the vSphere Client, it is hard to get an overview of how many snapshots have been created and how much space they are using, when they were created, and by whom they were created.

vSphere PowerCLI includes multiple cmdlets to allow you to work with snapshots. To view all snapshots on the current connection, type **Get-VM | Get-Snapshot**

To gain more information about all snapshots, type **Get-VM | Get-Snapshot | Select VM, Name, Description, Created, SizeMB | Format-Table**

In addition to reporting, PowerCLIvSphere PowerCLI also provides cmdlets for the management of snapshots.

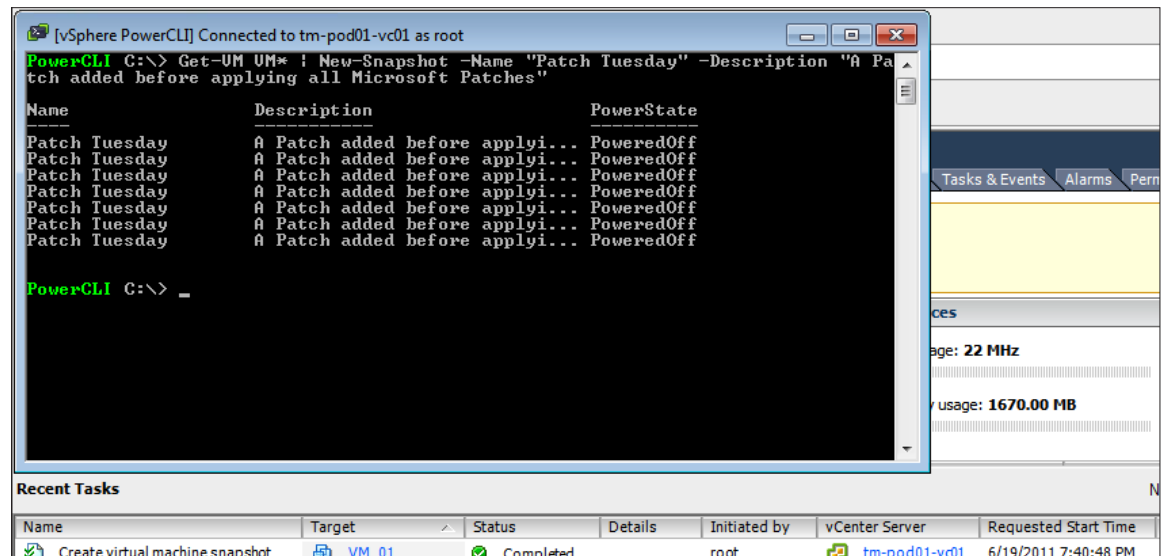


Figure 104.

Snapshots are easily created in large numbers in vSphere PowerCLI. Snapshots in the vSphere Client must be created one at a time. With vSphere PowerCLI, you can specify the criteria for your virtual machines and use the **New-Snapshot** cmdlet to create a snapshot on each virtual machine. The following example shows how to create a snapshot on all virtual machines having names that start with VM:

Get-VM VM* | New-Snapshot -Name "Patch Tuesday" -Description "A Patch added before applying all Microsoft Patches"

You will also see from the preceding screen shot that any task produced by vSphere PowerCLI will be recorded in the normal manner within vCenter and attributed to the user who is connected to this vSphere PowerCLI session.

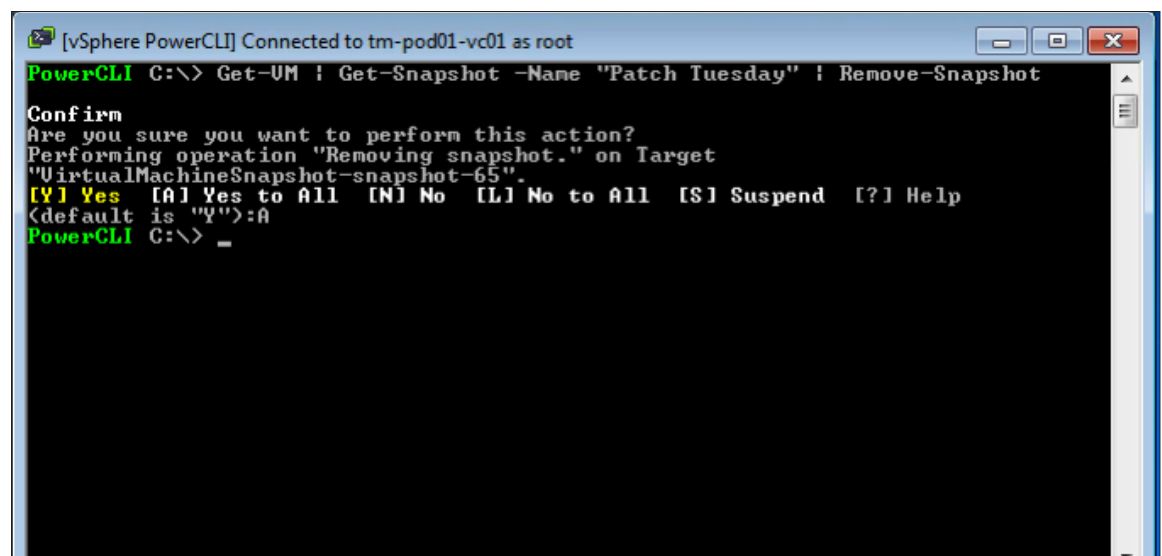


Figure 105.

As with the creation of snapshots, it is very easy to remove them in large numbers with the **Remove-Snapshot** cmdlet. The following example will remove all snapshots with a name of “Patch Tuesday”:

Get-Snapshot -Name “Patch Tuesday” | Remove-Snapshot

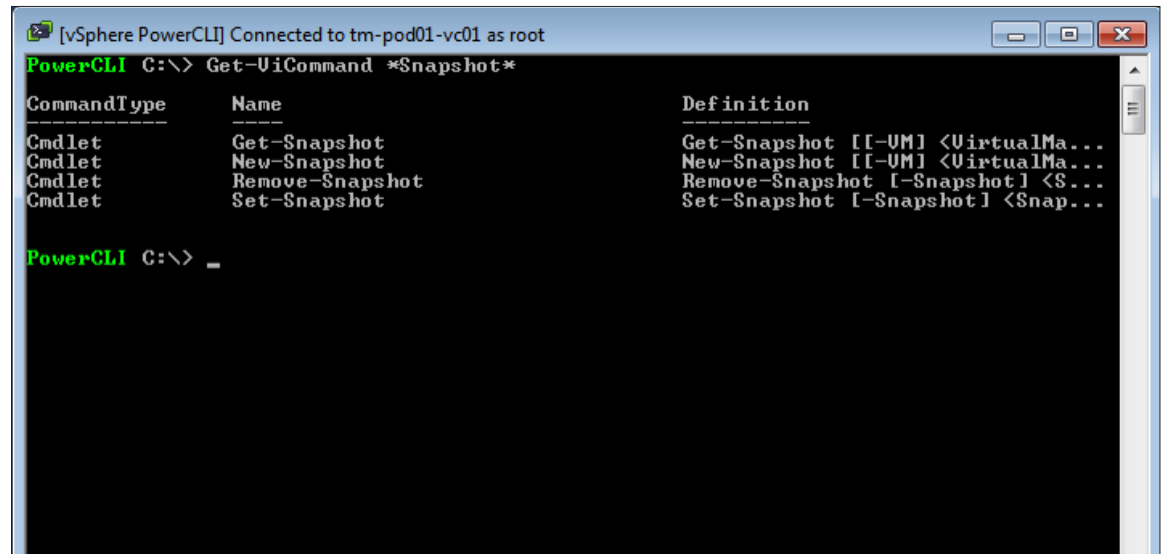


Figure 106.

To list the available cmdlets for working with snapshots, type **Get-ViCommand *Snapshot***

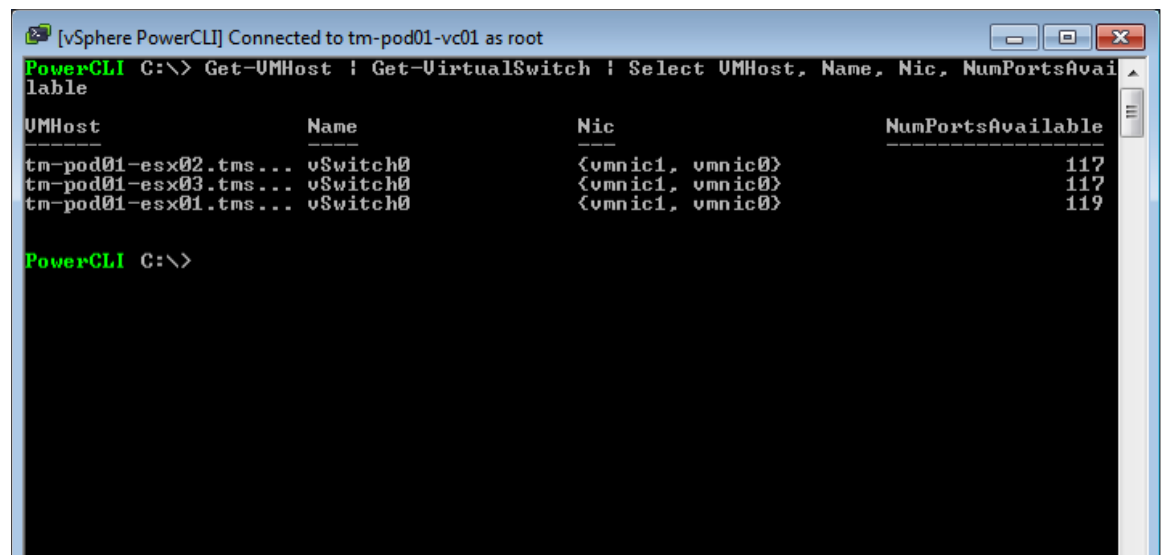
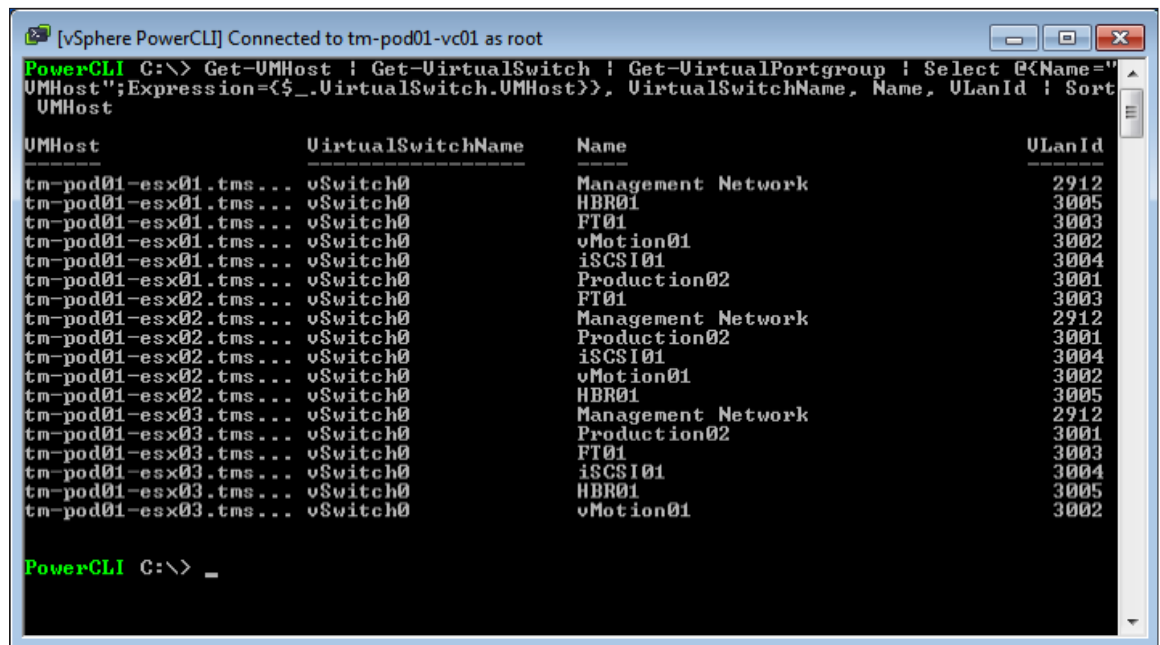


Figure 107.

Networking is also a key area of your virtual infrastructure. vSphere PowerCLI has the ability to report, create, and configure all aspects of your networking configuration.

To list all virtual switches and their information, type **Get-VMHost | Get-VirtualSwitch | Select VMHost, Name, Nic, NumPortsAvailable**



```
[vSphere PowerCLI] Connected to tm-pod01-vc01 as root
PowerCLI C:\> Get-VMHost | Get-VirtualSwitch | Get-VirtualPortgroup | Select @<Name="
VMHost";Expression={$_.VirtualSwitch.VMHost}>, VirtualSwitchName, Name, VlanId | Sort
VMHost
```

VMHost	VirtualSwitchName	Name	VlanId
tm-pod01-esx01.tms...	vSwitch0	Management Network	2912
tm-pod01-esx01.tms...	vSwitch0	HBR01	3005
tm-pod01-esx01.tms...	vSwitch0	FT01	3003
tm-pod01-esx01.tms...	vSwitch0	vMotion01	3002
tm-pod01-esx01.tms...	vSwitch0	iSCSI01	3004
tm-pod01-esx01.tms...	vSwitch0	Production02	3001
tm-pod01-esx02.tms...	vSwitch0	FT01	3003
tm-pod01-esx02.tms...	vSwitch0	Management Network	2912
tm-pod01-esx02.tms...	vSwitch0	Production02	3001
tm-pod01-esx02.tms...	vSwitch0	iSCSI01	3004
tm-pod01-esx02.tms...	vSwitch0	vMotion01	3002
tm-pod01-esx02.tms...	vSwitch0	HBR01	3005
tm-pod01-esx03.tms...	vSwitch0	Management Network	2912
tm-pod01-esx03.tms...	vSwitch0	Production02	3001
tm-pod01-esx03.tms...	vSwitch0	FT01	3003
tm-pod01-esx03.tms...	vSwitch0	iSCSI01	3004
tm-pod01-esx03.tms...	vSwitch0	HBR01	3005
tm-pod01-esx03.tms...	vSwitch0	vMotion01	3002

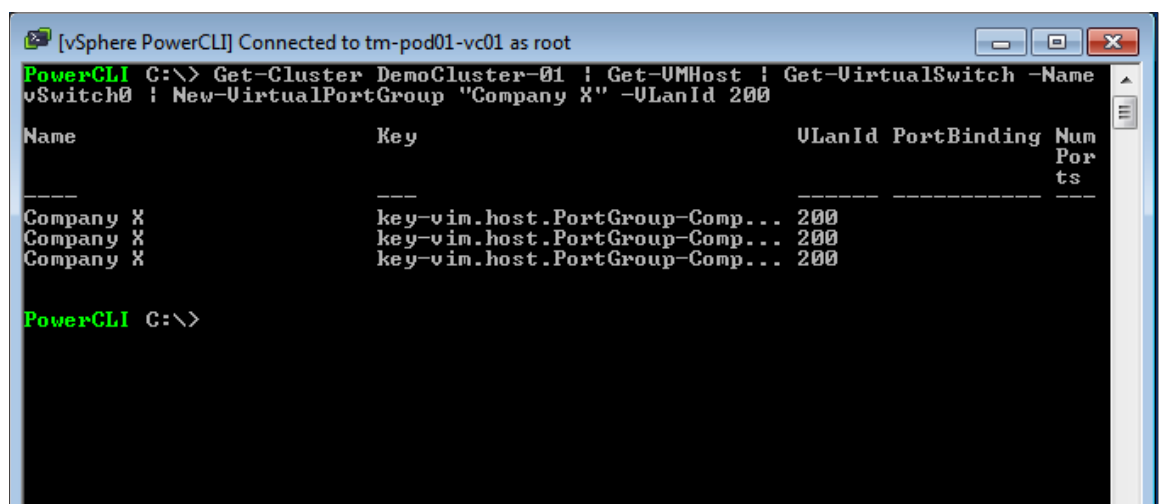
```
PowerCLI C:\> _
```

Figure 108.

vSphere PowerCLI reporting can also be used to ensure that your virtual configurations are correct. If a port group is missed, or the name is incorrect, or the VLANID has been set incorrectly, this can cause fundamental issues with clusters and the vSphere Distributed Resource Scheduler (DRS).

The preceding example shows how the configuration of each host, vSwitch, and port group can be checked. To perform this, type the following:

Get-VMHost | Get-VirtualSwitch | Get-VirtualPortgroup | Select @{Name="VMHost";Expression={\$_.VirtualSwitch.VMHost}}, VirtualSwitchName, Name, VlanId | Sort VMHost



```
[vSphere PowerCLI] Connected to tm-pod01-vc01 as root
PowerCLI C:\> Get-Cluster DemoCluster-01 | Get-VMHost | Get-VirtualSwitch -Name
vSwitch0 | New-VirtualPortGroup "Company X" -VlanId 200
```

Name	Key	VlanId	PortBinding	Num Por ts
Company X	key-vim.host.PortGroup-Comp...	200		
Company X	key-vim.host.PortGroup-Comp...	200		
Company X	key-vim.host.PortGroup-Comp...	200		

```
PowerCLI C:\>
```

Figure 109.

It is easy to add port groups in large numbers using vSphere PowerCLI. This can be achieved on each host in a specific cluster to ensure the DRS and HA compatibility of the host. The following example will create a new port

group called “Company X” on vSwitch0 for each host in the cluster “DemoCluster-01”:

Get-Cluster DemoCluster-01 | Get-VMHost | Get-VirtualSwitch -Name vSwitch0 | New-VirtualPortGroup “Company X” -VlanId 200

vSphere PowerCLI Summary

In conclusion, you can see that vSphere PowerCLI is a robust command-line tool for automating all aspects of vSphere management, including host, network, storage, virtual machine, and guest OS management. It can be used with other PowerShell snap-ins provided by Microsoft or third-party companies to integrate VMware technologies easily into other products and reach inside the guest OS.

The design of PowerShell and, inherently, vSphere PowerCLI, makes this scripting language easier to learn than many scripting languages before it. Complex configurations and reporting can be achieved with minimal effort from the administrator, safe in the knowledge of a repeatable, error-free solution.

Evaluating the ESXi Firewall

Introduction

The ESXi 5.0 management interface is protected by a service-oriented and stateless firewall, which you can configure using the vSphere Client or at the command line with `esxcli` interfaces. A new firewall engine eliminates the use of iptables, and rule sets define port rules for each service. For remote hosts, you can specify the IP addresses or range of IP addresses that are allowed to access each service.

Evaluation Overview

In this exercise, you will configure the ESXi firewall to allow or deny SSH service to the host. SSH is a service that can be enabled or stopped on an ESXi host. As part of this exercise, you will stop and start SSH service, and also configure firewall rules. ESXi firewall configuration can be done through the vSphere Client interface and through the vCLI. In this example environment, you will configure the firewall rules through vSphere Client UI.

Prerequisites

The evaluation environment consists of the following components:

1. Three ESXi hosts
2. Virtual machines running on hosts
3. Each virtual machine a software tool installed
 - a. PuTTY

Stopping SSH Service to Prevent Access

The SSH service provides a secure shell to manage the ESXi host. By default, this service is enabled. To stop this service, you have to follow these steps:

1. Select the **Home > Inventory > Hosts and Clusters** view.
2. Choose the host **tm-pod01-esx01.tmsb.local** in the left panel, and select **Configuration** tab on the right.
3. To see the firewall and services setting, select the **Security Profile** under the software section. Figure 110 shows the current Security Profile of the selected ESXi host. You can see that the SSH service is enabled and current firewall settings allow access to the SSH server on TCP port 22.

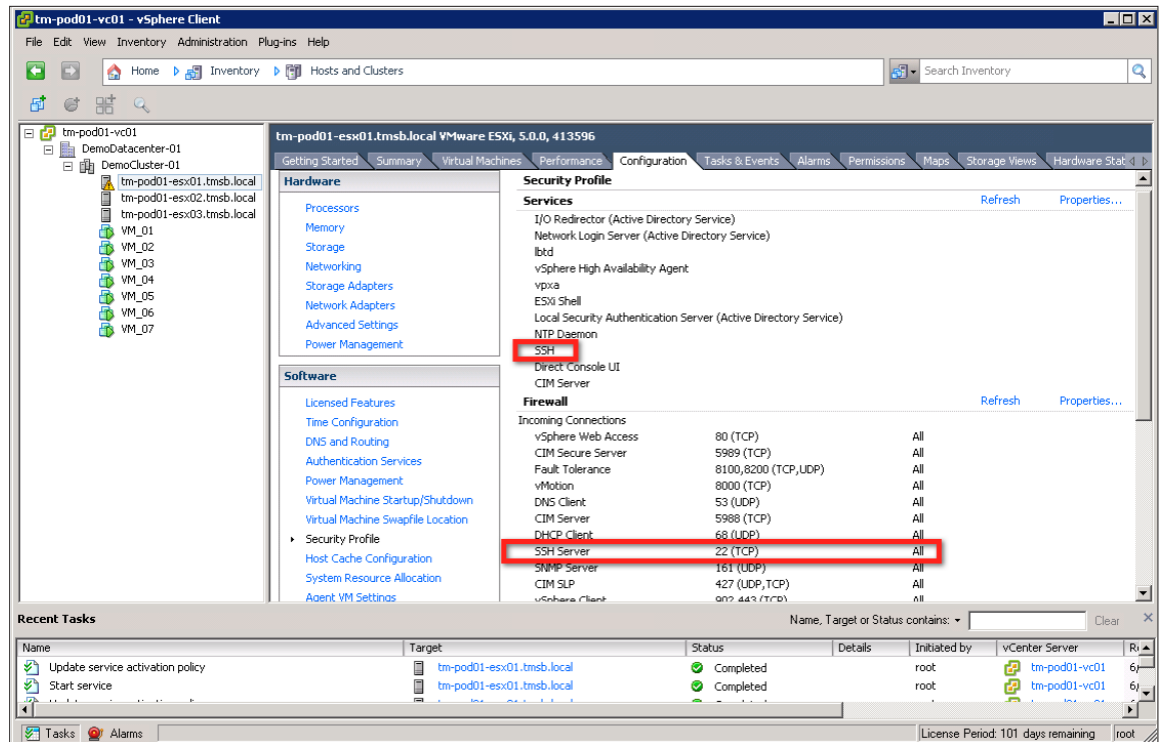


Figure 110.

- The warning sign on the host tm-pod01-esx01.tmsb.local is regarding the SSH service. Figure 111 shows the summary screen with the warning displayed. Enabling SSH service could be a security risk, so the platform provides the warning. You have to make sure that firewall rules are configured when SSH service is enabled.

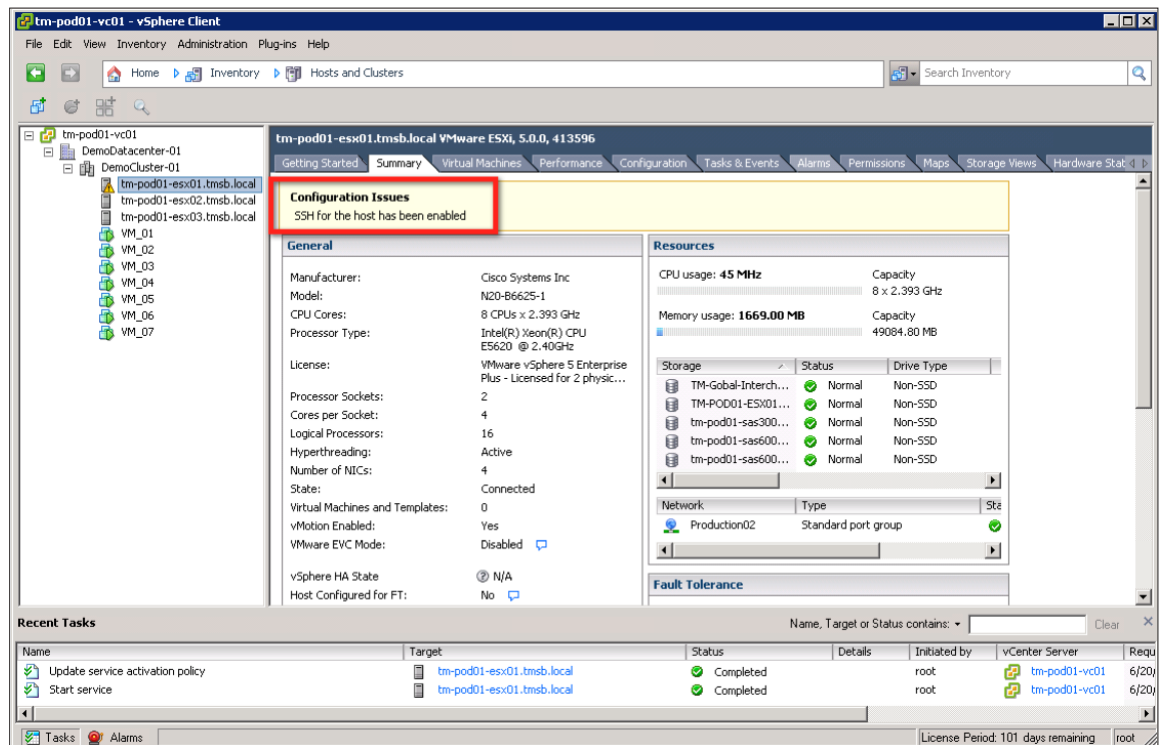


Figure 111.

5. To stop the service, you have to click the **Services Properties** link as shown in Figure 112.

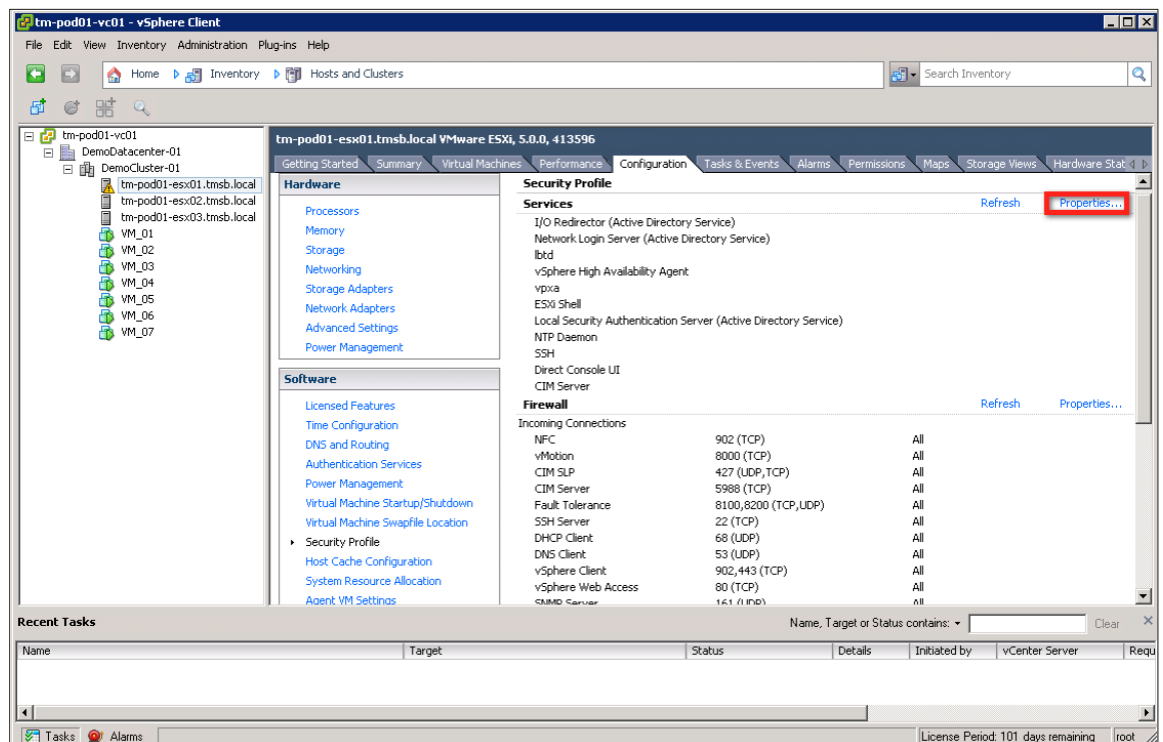


Figure 112.

6. This will bring up the panel shown in Figure 113. Select SSH and click **Options**. You can start or stop any services that are listed in this panel.

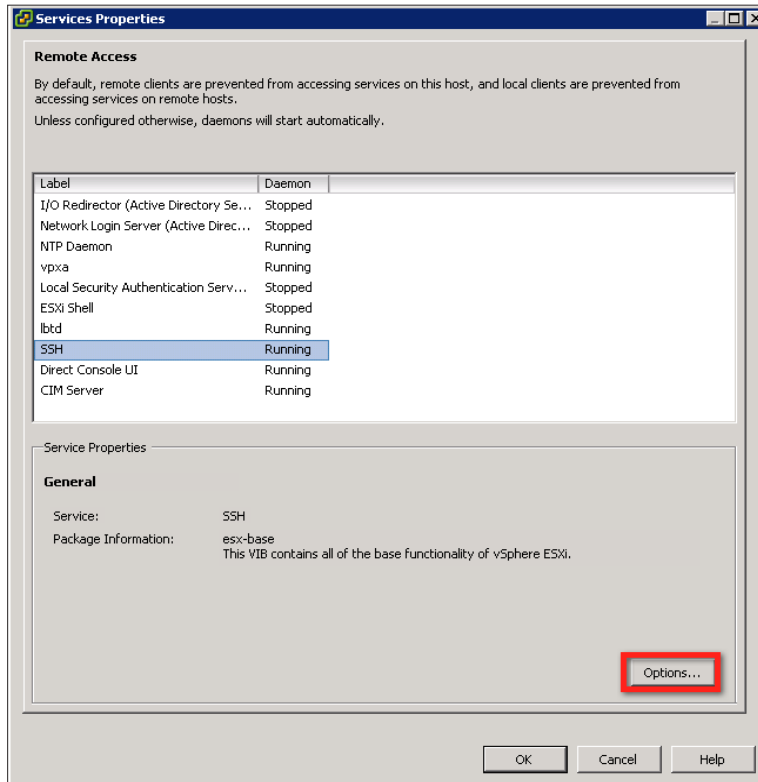


Figure 113.

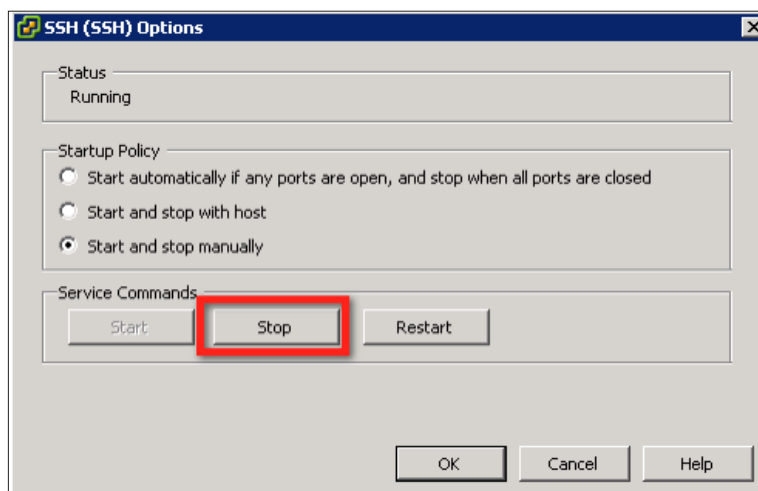


Figure 114.

7. Because this service was already started, you have an option to stop it by clicking Stop in the panel shown in Figure 114.

Testing Access with SSH Service Stopped

After stopping the SSH remote access service, you can test if any client can connect to Host1 (tm-pod01-esx01.tmsb.local) on TCP port 22.

In this example environment, you can use virtual machine VM_02 running on Host3 (tm-pod01-esx03.tmsb.local) to establish a SSH session with Host1. You can launch the PuTTY tool to establish the SSH session, as shown in Figure 115.

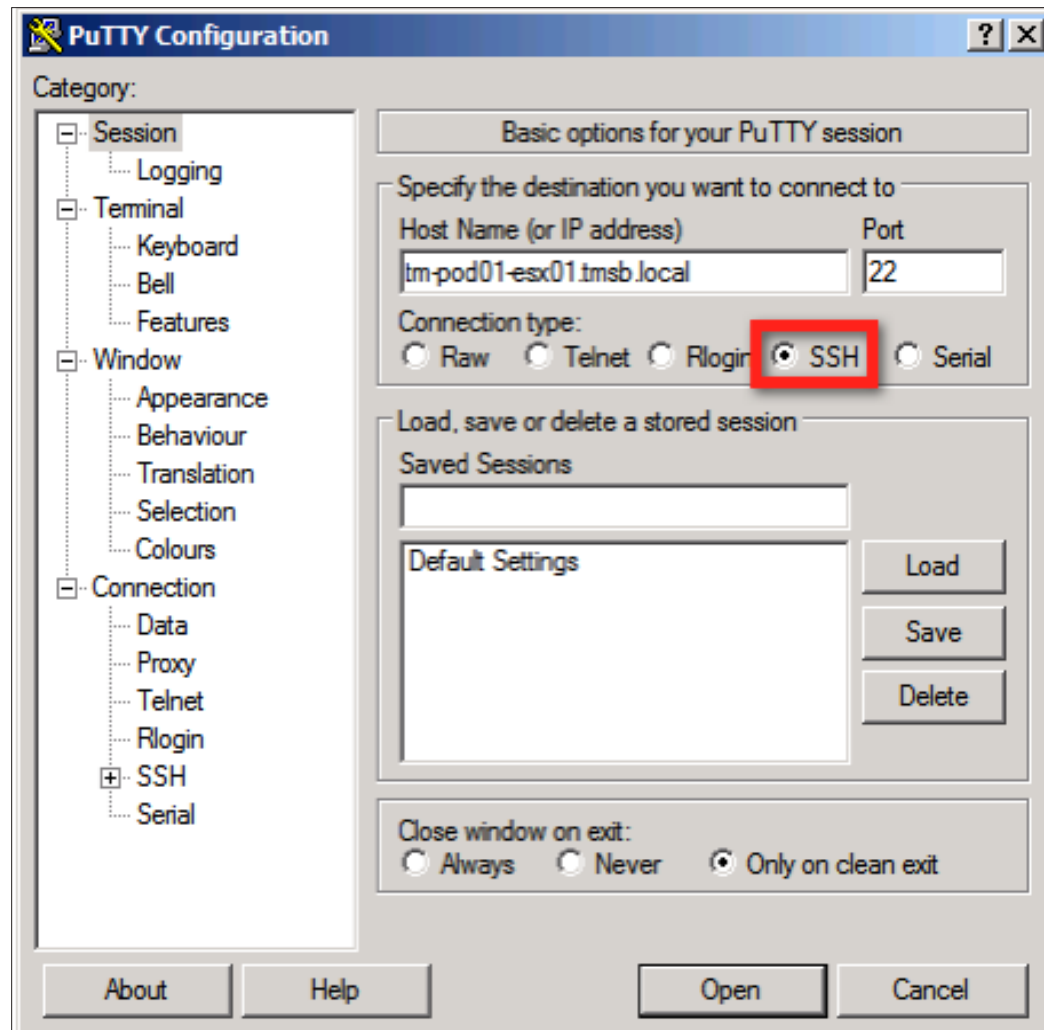


Figure 115.

The connection times out with a network error, as shown in Figure 116.

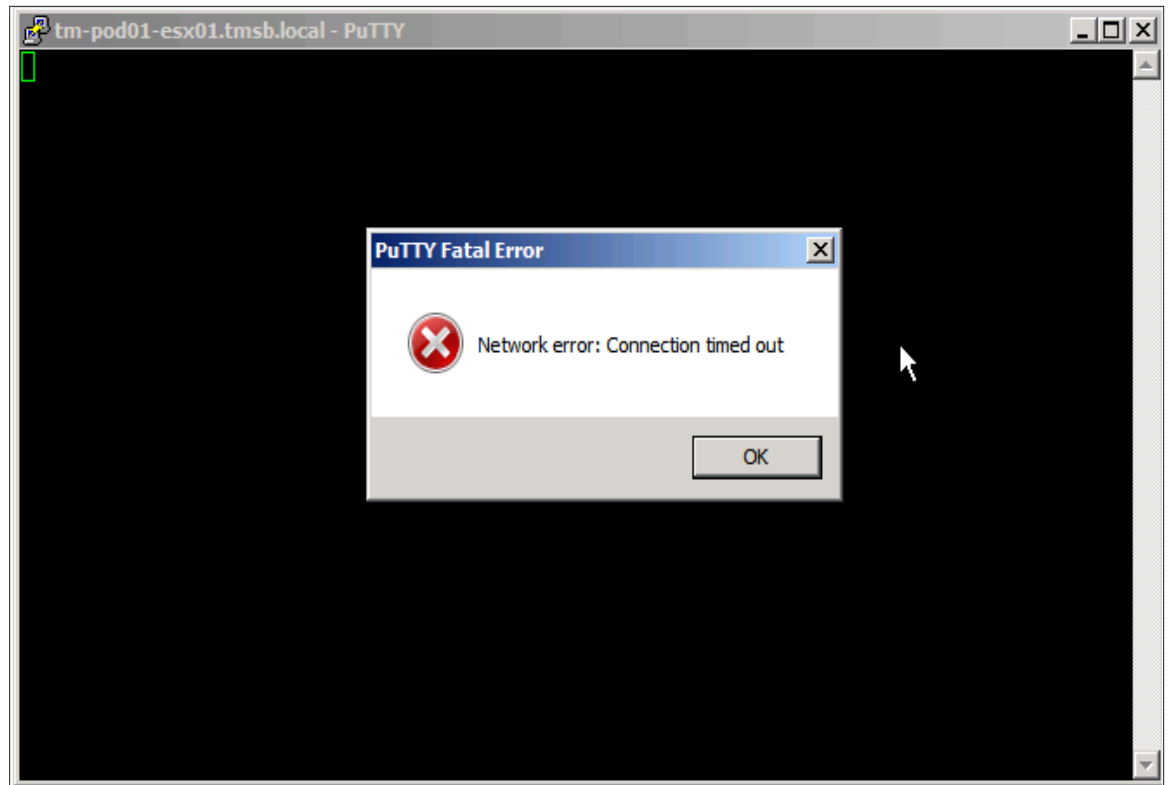


Figure 116.

This demonstrates that by shutting down the SSH service, you can completely deny remote access. Instead of blocking all access by stopping a service, you can selectively restrict remote access through the ESXi firewall. In the following section, you will enable the SSH service, and then use firewall settings to provide selective remote access.

Creating Firewall Rules to Block SSH Access

Before creating the firewall rules to block SSH access, you have to first enable the SSH service as follows:

1. Click the Services Properties link, as shown in Figure 112.
2. Select SSH service in the Service Properties panel, as shown in Figure 117. You can see that the SSH service is stopped. To enable the service, click **Options**.

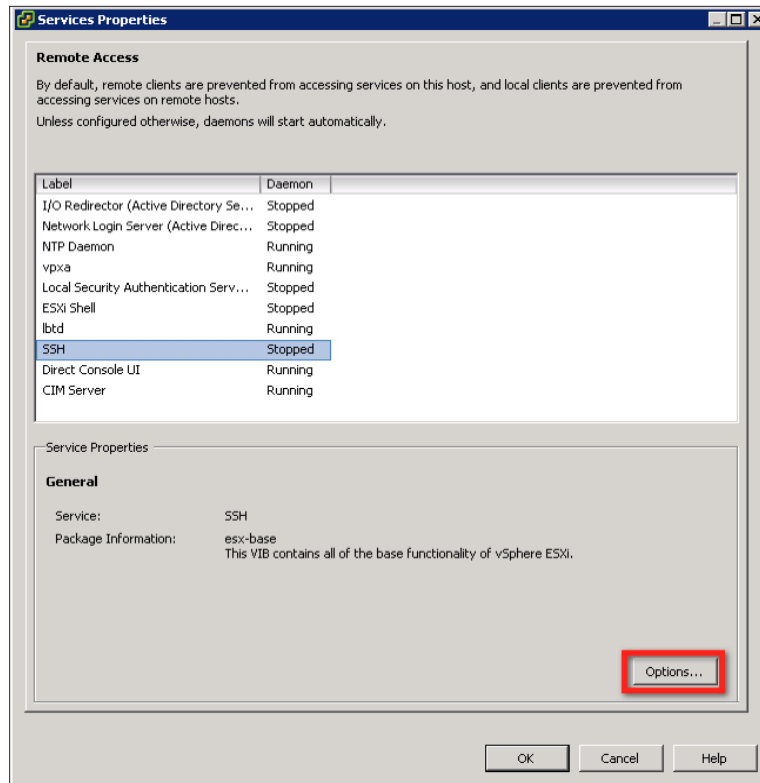


Figure 117.

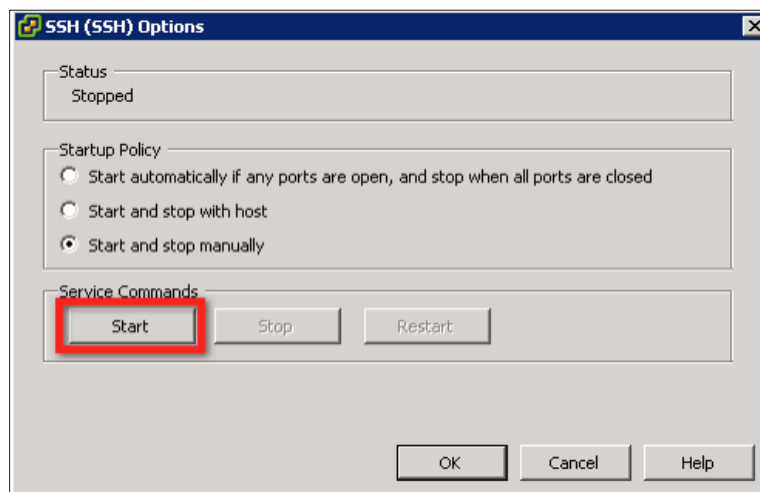


Figure 118.

- Click **Start** in the SSH Options panel, as shown in Figure 118. This will start the SSH service again. You can now configure the firewall rules for this service.
- Click firewall **Properties** to access the firewall setup panel. Figure 119 shows the firewall Properties.

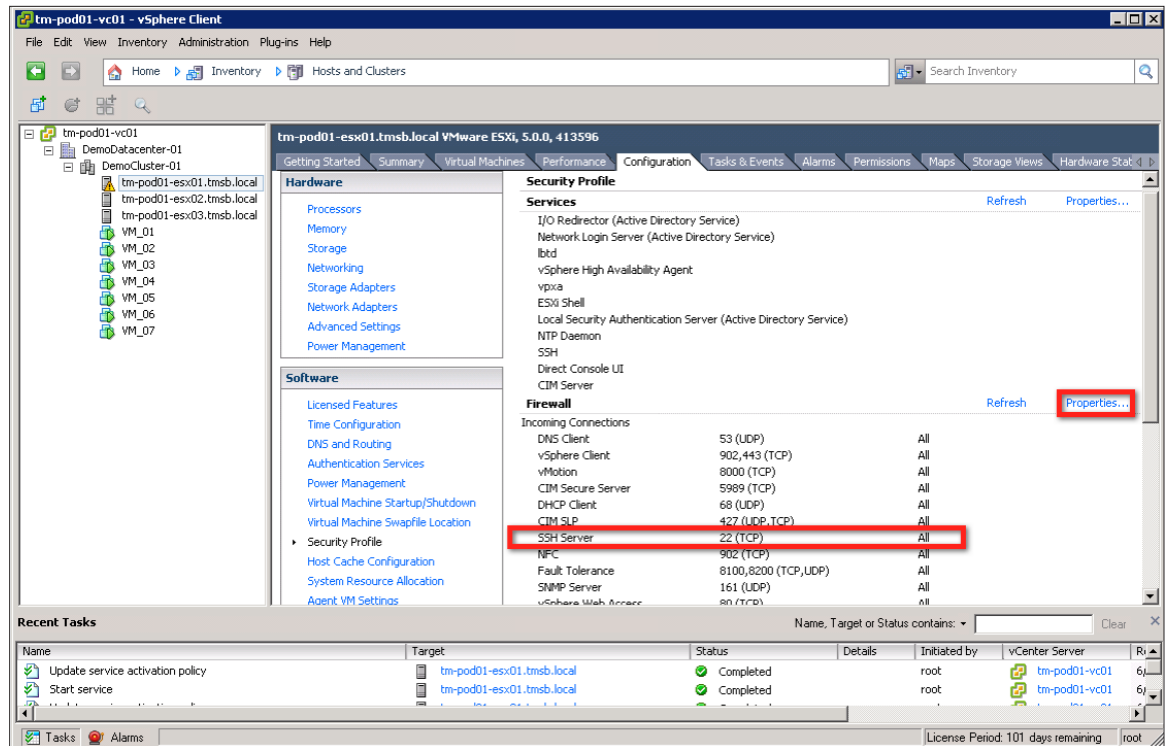


Figure 119.

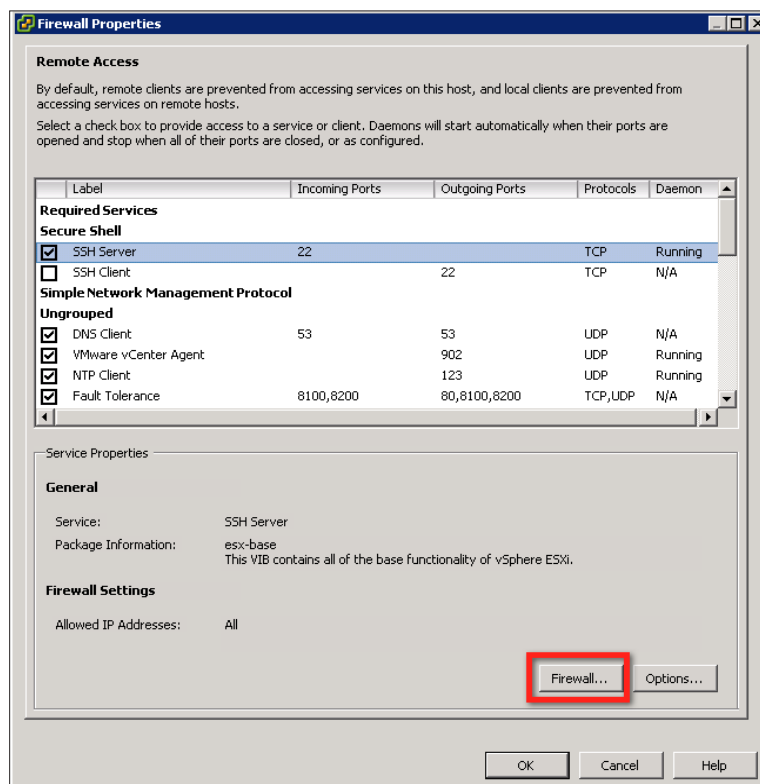


Figure 120.

5. After clicking the firewall Properties link, you will see the Firewall Properties panel, as shown in Figure 120. Select the SSH Server under the Secure Shell category, and click **Firewall**.
6. In this example environment, you have to enable the SSH remote access only from virtual machine VM_02 with IP address 10.91.35.55. SSH connections from all other IP addresses are denied. You can also give a range of IP addresses or subnet class in the “Only allow connections...” field shown in Figure 121.

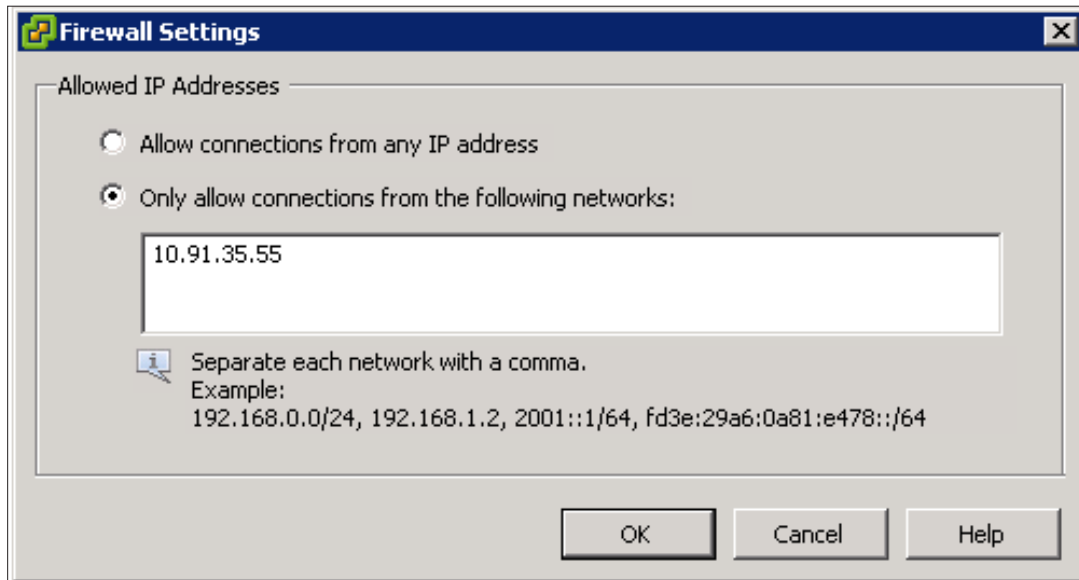


Figure 121.

After configuring the firewall rule to allow remote access for only virtual machine VM_02, you can test this firewall setting by establishing PuTTY sessions from different virtual machines.

Testing SSH Firewall Rules

In this example environment, you will try to establish SSH sessions from the following two virtual machines that are running from Host3 (tm-pod01-esx03.tmsb.local):

1. VM_02 : With IP address 10.91.35.55
2. VM_04 : With IP address 10.91.35.67

First, you can try creating a SSH session using the PuTTY tool on VM_02. This virtual machine IP address is one of the allowed IP addresses in the firewall configuration. Therefore, you can expect the SSH connection to be established.

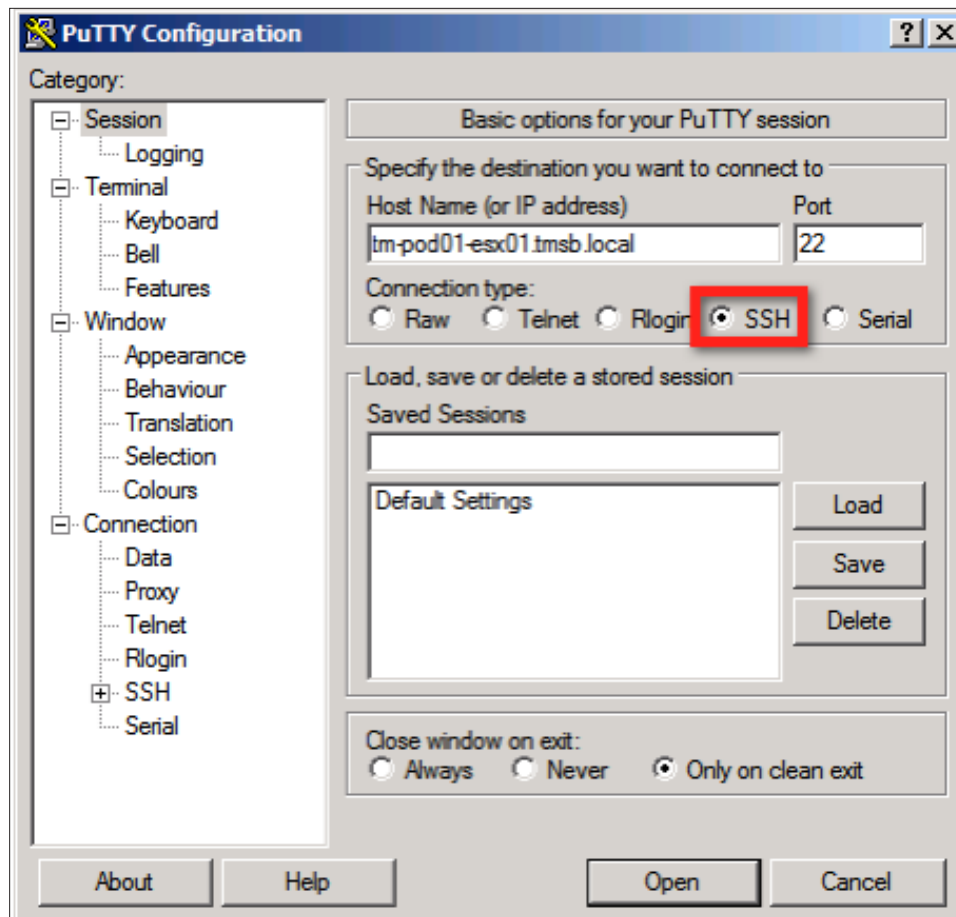


Figure 122.

Figure 123 shows the login screen of Host1. You can log in to the host with root credentials.

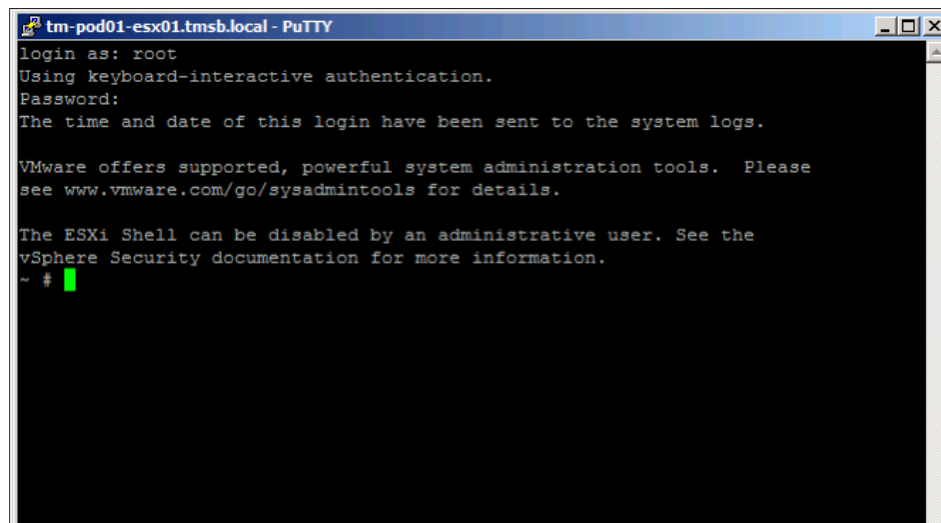


Figure 123.

When you repeat the step of establishing the SSH connection from VM_04 (10.91.35.67), you will get the “Network error: Connection timed out” message as shown in Figure 124. This is because the ESXi firewall blocks access on TCP port 22 from any IP address other than 10.91.35.55.

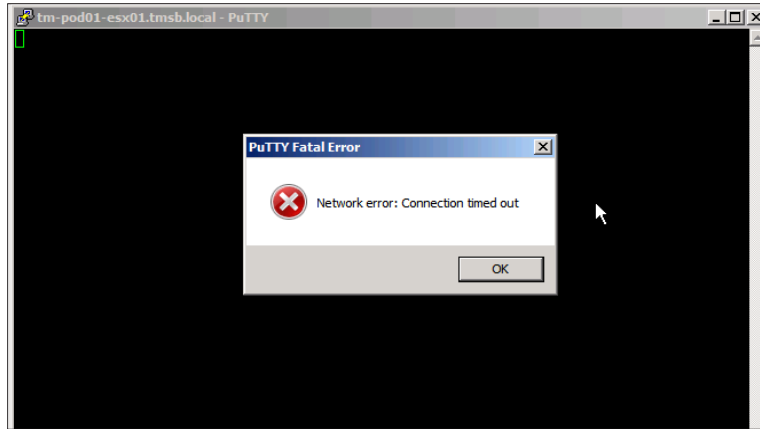


Figure 124.

Image Builder

Introduction

In this section, you will learn how to use vSphere 5.0 Image Builder to create and maintain custom ESXi images used to deploy hosts in your vSphere 5.0 environment. A past challenge with ESXi has been the static nature of the vSphere installation image. As customers adopt new hardware and as vendors release updates to CIM providers and software drivers, it was difficult to incorporate these updates into the ESXi installation. Image Builder enables users to update and maintain their ESXi images in order to keep up with the latest software drivers and updates.

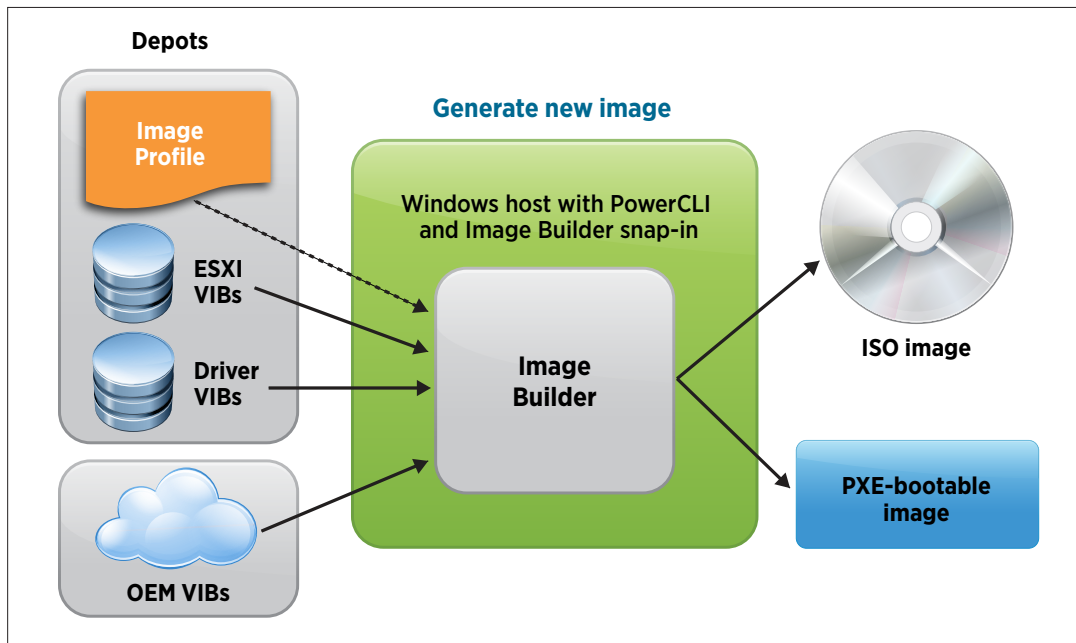


Figure 125. Image Builder Overview

Image Builder can be used in conjunction with VMware vSphere® Auto Deploy to dynamically provision hosts in a diskless environment.

Image Builder Prerequisites

The following components are required to use vSphere 5.0 Image Builder:

- Windows VM with 2GB of free disk space (used to host vSphere PowerCLI and store Image Builder software depots)
- vSphere PowerCLI 5.0
- ESXi Offline Bundle
- vCenter Server 5.0

Preparation Tasks

Complete the following steps prior to beginning your evaluation of Image Builder 5.0:

Install vSphere PowerCLI

Download and install vSphere 5.0 PowerCLI from www.vmware.com. The download file is a self-extracting executable file. Simply double-click on the .exe file to invoke the vSphere PowerCLI installer and follow the prompts. Refer to the *vSphere PowerCLI User's Guide* and the “vSphere PowerCLI by Example” section of this guide for more information on installing vSphere PowerCLI.

Download the ESXi Offline Bundle

Download the ESXi Offline Bundle ZIP file from www.vmware.com. The offline bundle is shipped in a ZIP format. Download the file on the same server where vSphere PowerCLI was installed.

Extract the ESXi Offline Bundle

Create the C:\ImageBuilder directory and extract the contents of the offline bundle into this directory.

Start an Image Builder vSphere PowerCLI Session

The following steps show how to start a vSphere PowerCLI session and how to connect to a vCenter Server.

Start vSphere PowerCLI by either double-clicking the vSphere PowerCLI icon on the desktop or selecting:

“Start -> Program -> VMware vSphere PowerCLI -> VMware vSphere PowerCLI”

From the vSphere PowerCLI prompt, run the “Connect-VIServer” cmdlet to connect your vSphere PowerCLI session to vCenter Server:

PowerCLI C:\> Connect-VIServer <vCenter IP address>

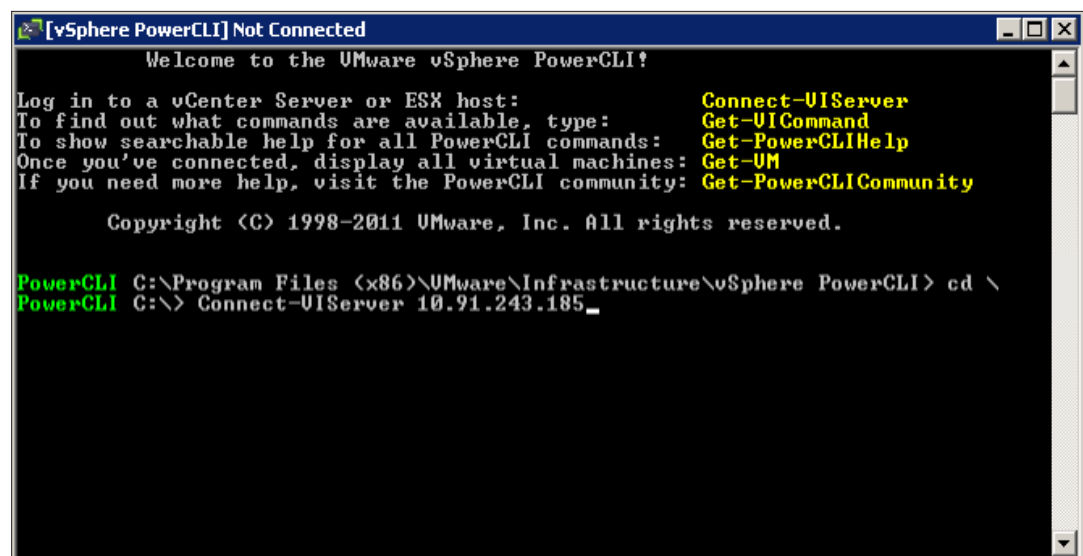


Figure 126. Connect-VIServer

Depending on your login credentials, you might be prompted to enter the vCenter user name and password, as follows:

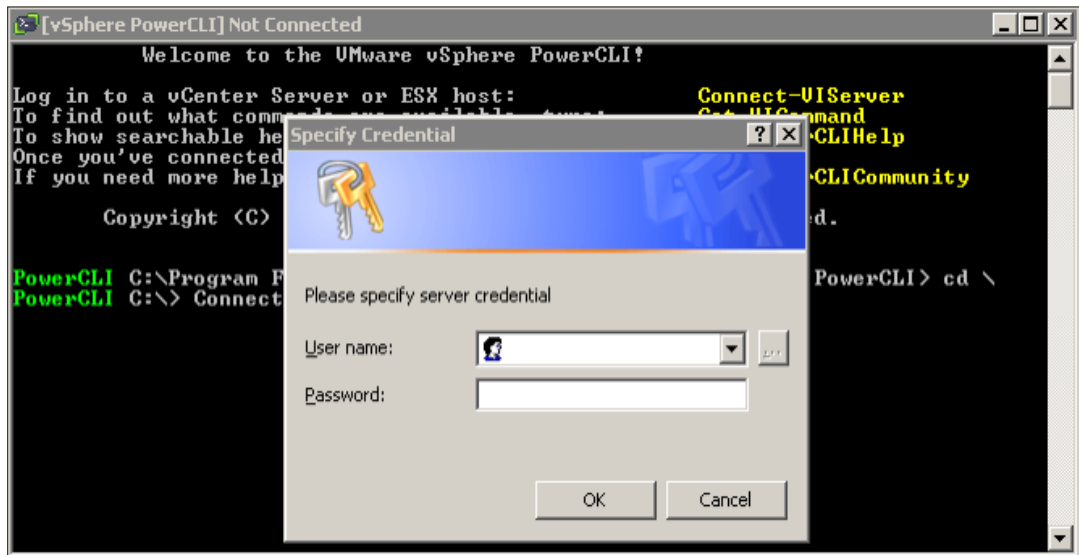


Figure 127. Connect-VIServer Login Prompt

vSphere PowerCLI will show the vCenter Server name/IP and the port and user. During the Image Builder evaluation, the certificate error can be ignored.

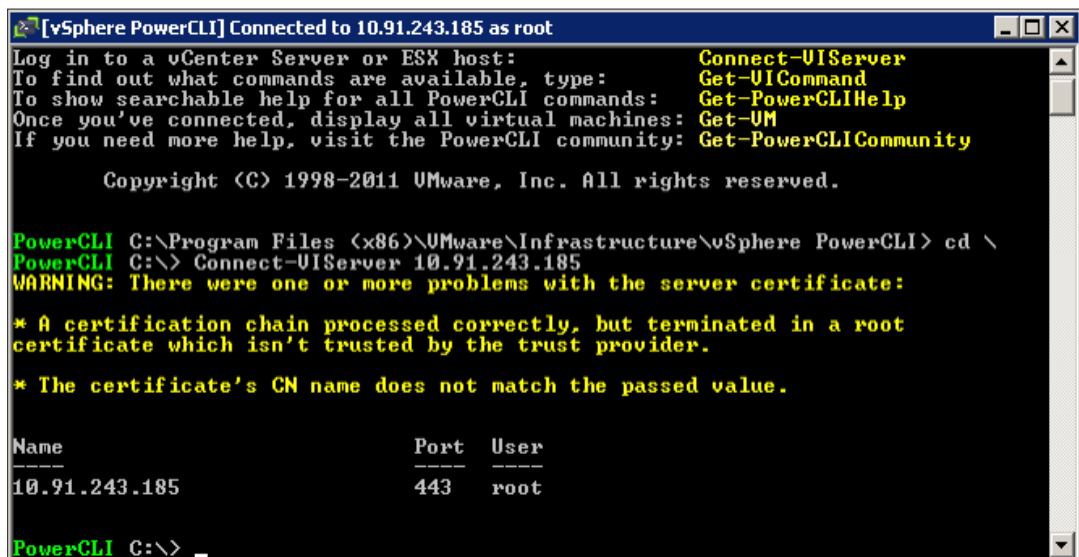
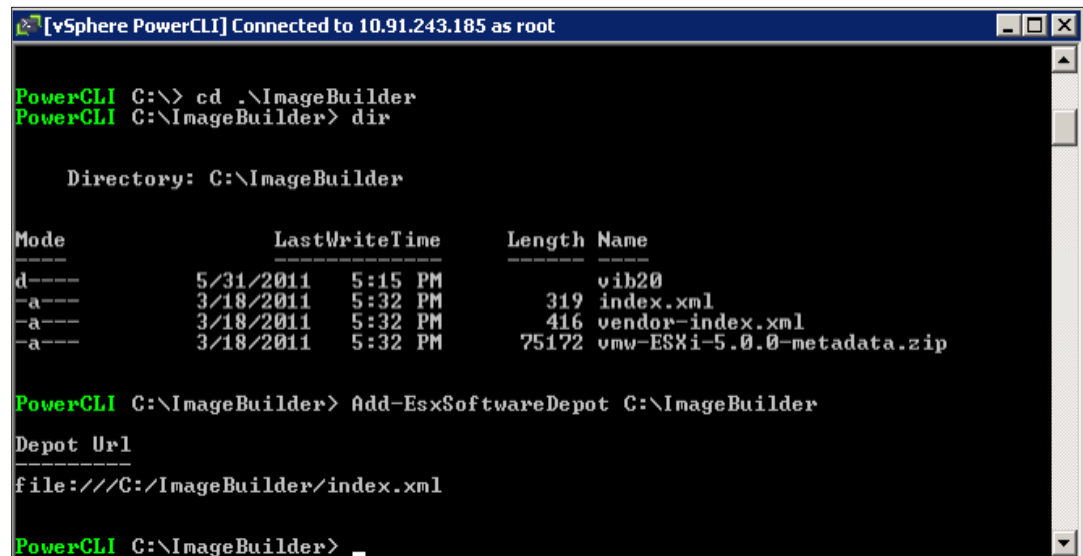


Figure 128. Connect-VIServer Results

Import the ESXi Offline Bundle

This section shows how to import an ESXi software depot using the ESXi offline depot staged in the C:\ImageBuilder directory during the preparation tasks.

PowerCLI C:\> Add-EsxSoftwareDepot C:\ImageBuilder



```

[vSphere PowerCLI] Connected to 10.91.243.185 as root

PowerCLI C:\> cd .\ImageBuilder
PowerCLI C:\ImageBuilder> dir

    Directory: C:\ImageBuilder

Mode                LastWriteTime         Length Name
----                -
d-----          5/31/2011   5:15 PM             vib20
-a----          3/18/2011   5:32 PM             319 index.xml
-a----          3/18/2011   5:32 PM             416 vendor-index.xml
-a----          3/18/2011   5:32 PM        75172 vmw-ESXi-5.0.0-metadata.zip

PowerCLI C:\ImageBuilder> Add-EsxSoftwareDepot C:\ImageBuilder

Depot Url
-----
file:///C:/ImageBuilder/index.xml

PowerCLI C:\ImageBuilder> _
  
```

Figure 129. Add Software Depot

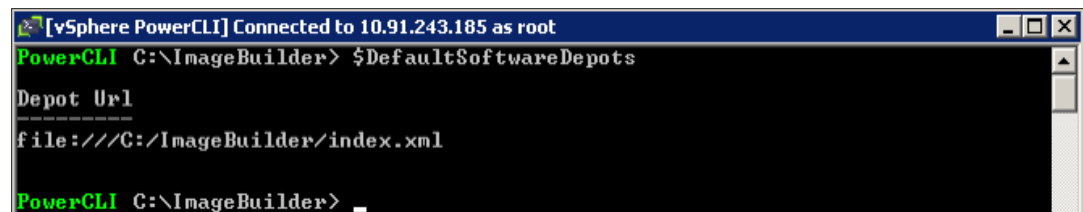
The software depot is a collection of vSphere packages used to create and maintain ESXi images. The following steps show how to view information about the software depots added to your vSphere PowerCLI session.

Display Software Depots

Software depots are added using the **Add-ESXSoftwareDepot** cmdlet and removed using the **Remove-SoftwareDepot** cmdlet. Use the **\$DefaultSoftwareDepots** variable to view the list of software depots available in your current vSphere PowerCLI session.

To view available software depots, type the following:

PowerCLI C:\> \$DefaultSoftwareDepots



```

[vSphere PowerCLI] Connected to 10.91.243.185 as root

PowerCLI C:\ImageBuilder> $DefaultSoftwareDepots

Depot Url
-----
file:///C:/ImageBuilder/index.xml

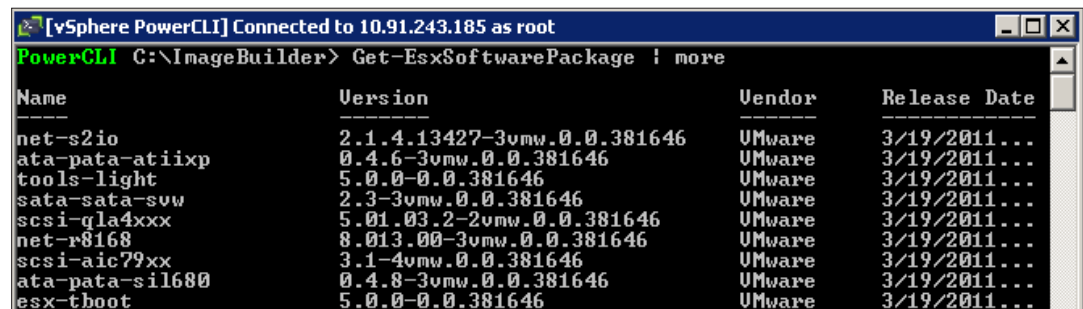
PowerCLI C:\ImageBuilder> _
  
```

Figure 130. Display Software Depot

Display VIBs

A vSphere Installation Bundle (VIB) is a packaging format used in vSphere. VMware and its partners package solutions, drivers, CIM providers and applications as VIBs. VIBs are then grouped together to create ESXi image profiles. To view the available VIBs from the software depots added to your vSphere PowerCLI session, use the **Get-EsxSoftwarePackage** cmdlet.

PowerCLI C:\> Get-EsxSoftwarePackage



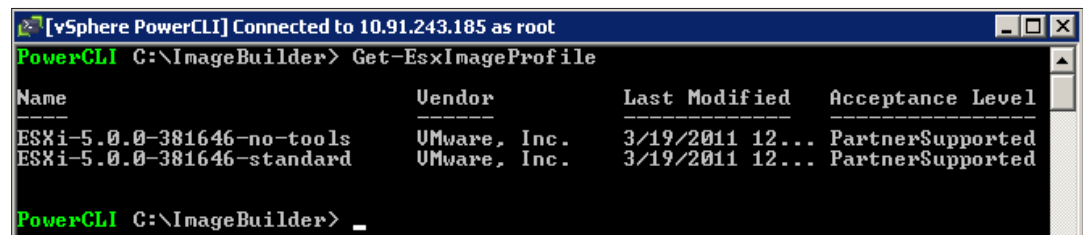
Name	Version	Vendor	Release Date
net-s2io	2.1.4.13427-3vmw.0.0.381646	VMware	3/19/2011...
ata-pata-atiixp	0.4.6-3vmw.0.0.381646	VMware	3/19/2011...
tools-light	5.0.0-0.0.381646	VMware	3/19/2011...
sata-sata-svw	2.3-3vmw.0.0.381646	VMware	3/19/2011...
scsi-gla4xxx	5.01.03.2-2vmw.0.0.381646	VMware	3/19/2011...
net-r8168	8.013.00-3vmw.0.0.381646	VMware	3/19/2011...
scsi-aic79xx	3.1-4vmw.0.0.381646	VMware	3/19/2011...
ata-pata-sil680	0.4.8-3vmw.0.0.381646	VMware	3/19/2011...
esx-tboot	5.0.0-0.0.381646	VMware	3/19/2011...

Figure 131. Get-EsxSoftwarePackage

Display Image Profiles

An image profile is a compilation of VIBs that make up an ESXi image that can be used to install an ESXi host. At a minimum, an image profile is comprised of a base ESXi VIB and a bootable kernel module VIB, but can also include additional VIBs from the pool of available software depots. To list the configured image profiles, use the **Get-EsxImageProfile** cmdlet.

PowerCLI C:\> Get-EsxImageProfile

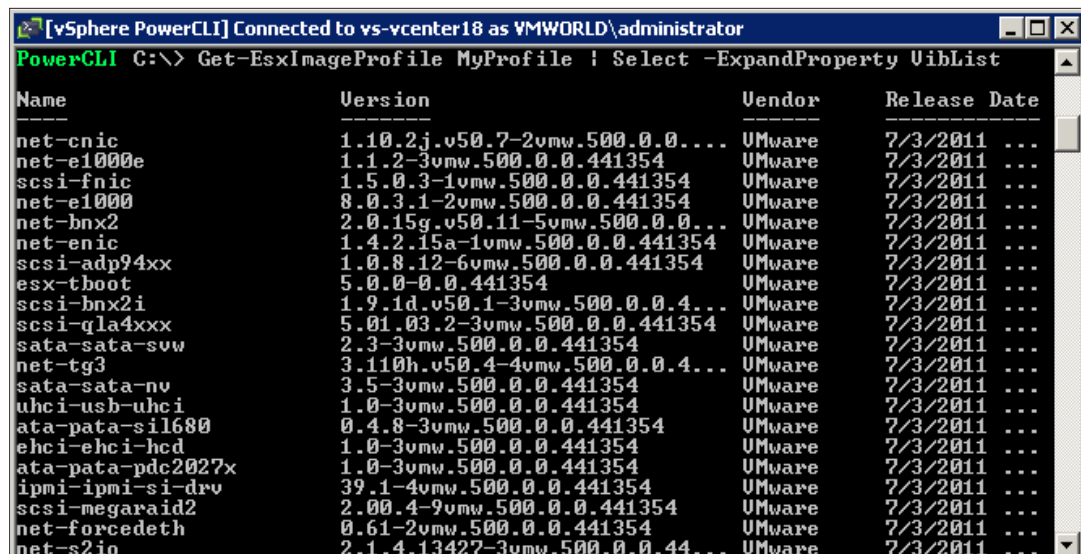


Name	Vendor	Last Modified	Acceptance Level
ESXi-5.0.0-381646-no-tools	VMware, Inc.	3/19/2011 12...	PartnerSupported
ESXi-5.0.0-381646-standard	VMware, Inc.	3/19/2011 12...	PartnerSupported

Figure 132. Get-EsxImageProfile

To list the VIBs that comprise an image profile, use the **Get-EsxImageProfile** cmdlet and expand the properties of the **VibList** property.

PowerCLI C:\> Get-EsxImageProfile MyProfile | Select -ExpandProperty VibList



```
[vSphere PowerCLI] Connected to vs-vcenter18 as VMWORLD\administrator
PowerCLI C:\> Get-EsxImageProfile MyProfile | Select -ExpandProperty VibList
```

Name	Version	Vendor	Release Date
net-cnic	1.10.2j.v50.7-2vmw.500.0.0...	VMware	7/3/2011 ...
net-e1000e	1.1.2-3vmw.500.0.0.441354	VMware	7/3/2011 ...
scsi-fnic	1.5.0.3-1vmw.500.0.0.441354	VMware	7/3/2011 ...
net-e1000	8.0.3.1-2vmw.500.0.0.441354	VMware	7/3/2011 ...
net-bnx2	2.0.15g.v50.11-5vmw.500.0.0...	VMware	7/3/2011 ...
net-enic	1.4.2.15a-1vmw.500.0.0.441354	VMware	7/3/2011 ...
scsi-adp94xx	1.0.8.12-6vmw.500.0.0.441354	VMware	7/3/2011 ...
esx-tboot	5.0.0-0.0.441354	VMware	7/3/2011 ...
scsi-bnx2i	1.9.1d.v50.1-3vmw.500.0.0.4...	VMware	7/3/2011 ...
scsi-qla4xxx	5.01.03.2-3vmw.500.0.0.441354	VMware	7/3/2011 ...
sata-sata-svw	2.3-3vmw.500.0.0.441354	VMware	7/3/2011 ...
net-tg3	3.110h.v50.4-4vmw.500.0.0.4...	VMware	7/3/2011 ...
sata-sata-nv	3.5-3vmw.500.0.0.441354	VMware	7/3/2011 ...
uhci-usb-uhci	1.0-3vmw.500.0.0.441354	VMware	7/3/2011 ...
ata-pata-sil680	0.4.8-3vmw.500.0.0.441354	VMware	7/3/2011 ...
ehci-ehci-hcd	1.0-3vmw.500.0.0.441354	VMware	7/3/2011 ...
ata-pata-pdc2027x	1.0-3vmw.500.0.0.441354	VMware	7/3/2011 ...
ipmi-ipmi-si-drv	39.1-4vmw.500.0.0.441354	VMware	7/3/2011 ...
scsi-megaraid2	2.00.4-9vmw.500.0.0.441354	VMware	7/3/2011 ...
net-forcedeth	0.61-2vmw.500.0.0.441354	VMware	7/3/2011 ...
net-s2io	2.1.4.13427-3vmw.500.0.0.44...	VMware	7/3/2011 ...

Figure 133. Get-EsxImageProfile VibList

Create a New Image Profile

The following steps show how to create a custom image profile either by manually selecting the individual VIB components or by cloning an existing image profile.

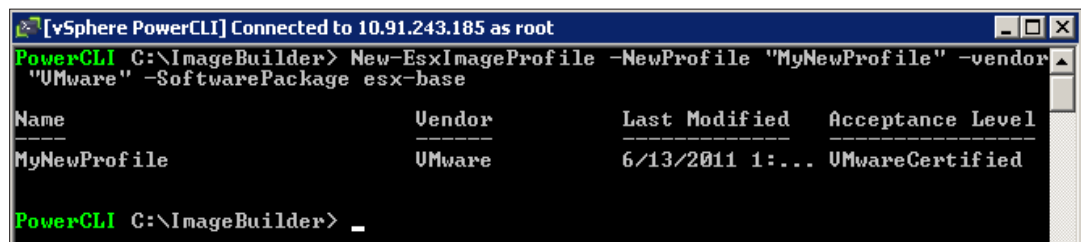
Each image profile must meet the following requirements:

- The image profile must have a unique name.
- The image profile must contain at least one base ESXi VIB and one bootable kernel module.
- The acceptance level for each VIB must match the acceptance level defined for the profile.
- A VIB can only exist once in an image profile.
- All VIB dependencies must be met.

Create a New Image Profile by Manually Selecting Individual VIBs

Create a new image profile named **"MyNewProfile"** that contains the ESXi base image.

```
PowerCLI C:\> New-EsxImageProfile -NewProfile "MyNewProfile" -vendor "VMware" -
SoftwarePackage esx-base
```



```
[vSphere PowerCLI] Connected to 10.91.243.185 as root
PowerCLI C:\ImageBuilder> New-EsxImageProfile -NewProfile "MyNewProfile" -vendor
"VMware" -SoftwarePackage esx-base
```

Name	Vendor	Last Modified	Acceptance Level
MyNewProfile	VMware	6/13/2011 1:...	VMwareCertified

```
PowerCLI C:\ImageBuilder> _
```

Figure 134. New-EsxImageProfile

Next, add the VIB “**esx-tboot**” to “**MyNewProfile**” as follows:

PowerCLI C:\> Add-EsxSoftwarePackage -ImageProfile “MyNewProfile” -SoftwarePackage “esx-tboot”

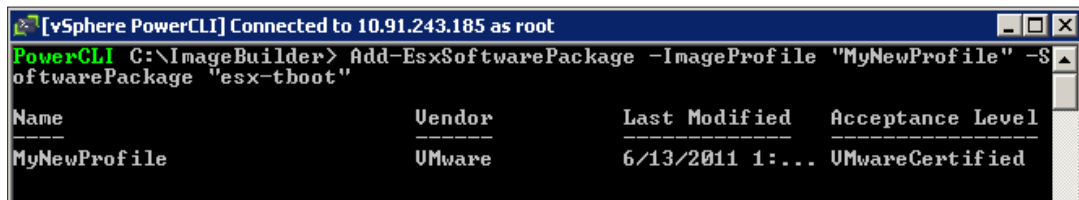


Figure 135. Add-EsxSoftwarePackage esx-tboot

Next, add the VIB “**net-e1000e**” to “**MyNewProfile**” as follows:

PowerCLI C:\> Add-EsxSoftwarePackage -ImageProfile “MyNewProfile” -SoftwarePackage “net-e1000e”

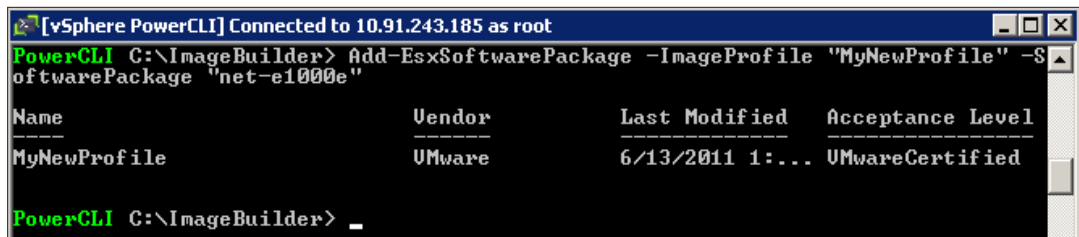


Figure 136. Add-EsxSoftwarePackage net-e1000e

Next, display the available image profiles and confirm that the new image profile “**MyNewProfile**” has been created:

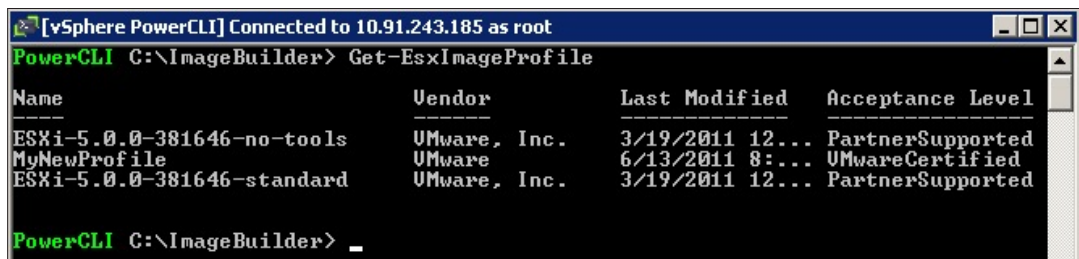


Figure 137. Get-EsxImageProfile with MyNewProfile

Next, display the list of VIBs in the image profiles to confirm that only the VIBs identified are included:

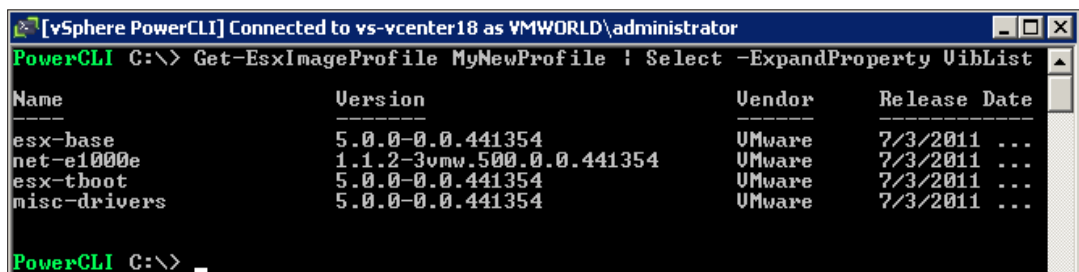


Figure 138. Get-EsxImageProfile MyNewProfile VibList

Create a New Image Profile by Cloning an Existing Image Profile

Create a new ESXi image named “**MyClonedProfile**” by cloning the **ESXi-5.0.0-381646-standard Image** included with the offline bundle.

PowerCLI C:\> New-EsxImageProfile -CloneProfile ESXi-5.0.0-381646-standard -Name “MyClonedProfile”

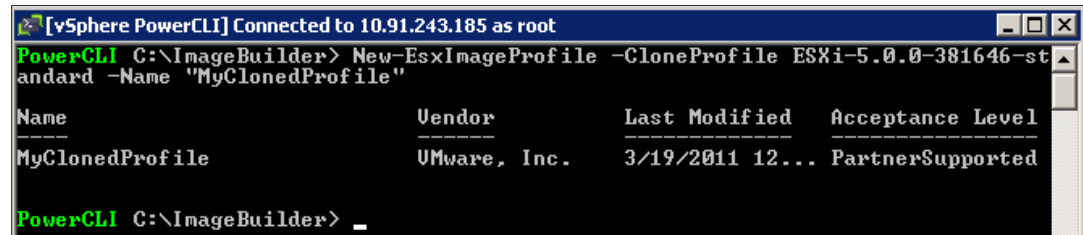


Figure 139. New-EsxImageProfile -CloneProfile

Display the list of available image profiles confirming that the new profile was created:

PowerCLI C:\> Get-EsxImageProfile

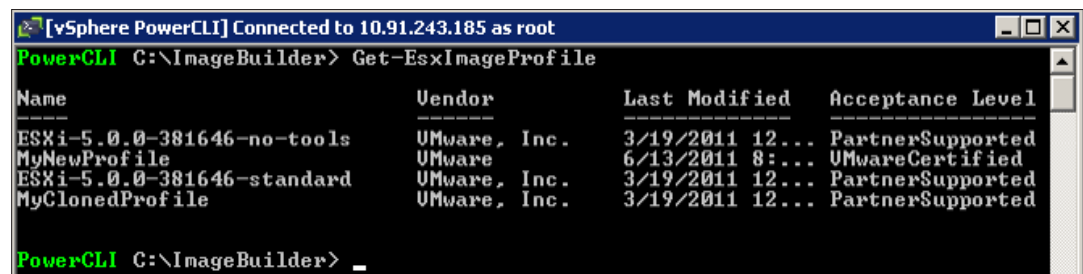


Figure 140. Get-EsxImageProfile with MyClonedProfile

Removing VIBs from an Image Profile

The cloned image profile “**MyCloneProfile**” includes the VMware Tools package. We can make the size of this image profile smaller by removing the VMware Tools package.

PowerCLI C:\> Remove-EsxSoftwarePackage -ImageProfile MyClonedProfile -SoftwarePackage tools-light

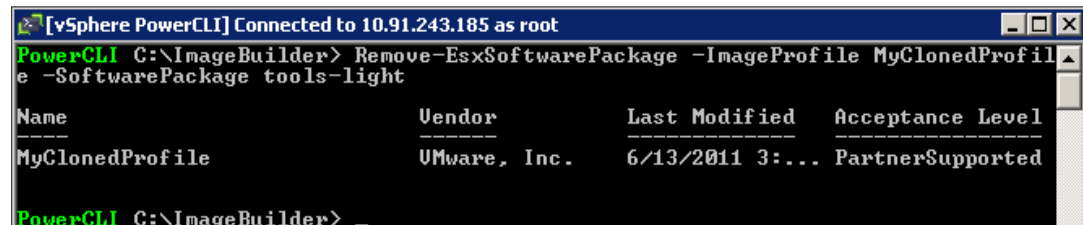


Figure 141. Remove-EsxSoftwarePackage

Compare Image Profiles

This section shows how to compare image profiles to help identify and track differences between custom image profiles.

In the previous section, we created a clone of the default image profile called “**MyCloneProfile**”. We then removed the VMware Tools package from the custom image. We can now use the **Compare-EsxImageProfile** cmdlet to compare the two images and verify the changes that were made.

```
PowerCLI C:\> Compare-EsxImageProfile -ReferenceProfile Esxi-5.0.0-381646-no-tools
-CompareProfile MyClonedProfile
```

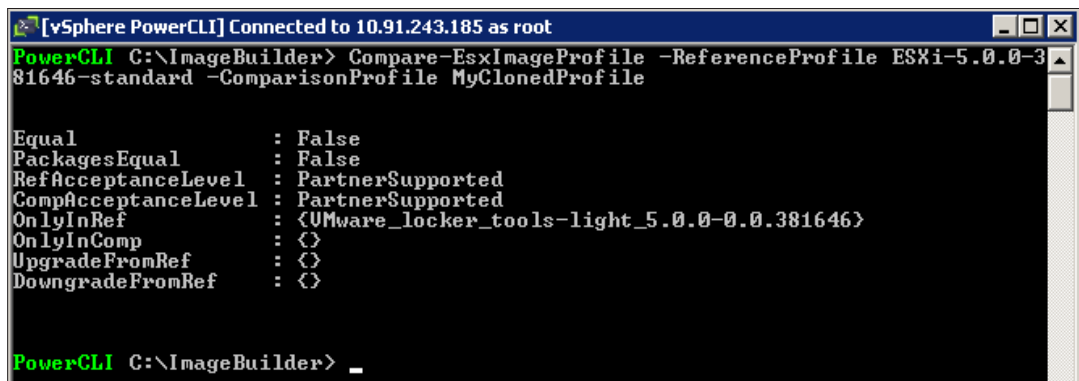


Figure 142. Compare-EsxImageProfile

In the preceding example, we can see that the package **VMware_locker_tools-light_5.0.0-0.0.381646** does not exist in the reference profile (-ReferenceProfile) but does exist in the comparison profile (-Comparison Profile).

Export Image Profile

The following steps show how to export image profiles as an offline bundle or as a bootable ISO image.

Export As an Offline Bundle

Each time you exit your vSphere PowerCLI session, all software depots and custom image profiles are lost. To save your custom image profiles, in order to continue to work with them between vSphere PowerCLI sessions, you must save them to disk by exporting to an offline bundle. With an offline bundle, each time you start a new vSphere PowerCLI session, you can continue to work with your custom image profiles by importing the offline bundle as a new software depot using the **Add-EsxSoftwareDepot** cmdlet.

To export an image profile as an offline bundle, use the **Export-EsxImageProfile** cmdlet with the **-ExportToBundle** option.

```
PowerCLI C:\> Export-EsxImageProfile -ImageProfile MyNewProfile -ExportToBundle -
FilePath C:\ImageBuilder\MyNewProfile
```

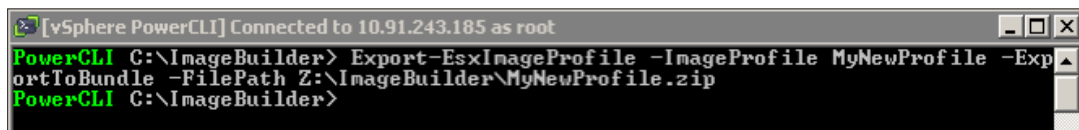


Figure 143. Export-EsxImageProfile -ExportToBundle

Export As a Bootable ISO Image

In order to use a custom image profile to install ESXi hosts, you must export the image profile as a bootable ISO. Use the **Export-EsxImageProfile** cmdlet with the **-ExportToIso** option.

```
PowerCLI C:\> Export-EsxImageProfile -ImageProfile MyNewProfile -ExportToIso -
FilePath C:\ImageBuilder\MyNewProfile.iso
```

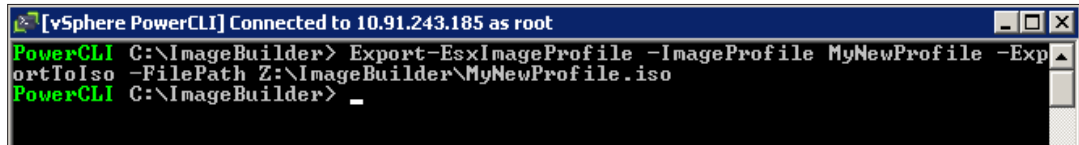


Figure 144. Export-EsxImageProfile -ExportToIso

Use Windows Explorer to view the ZIP and .iso files.

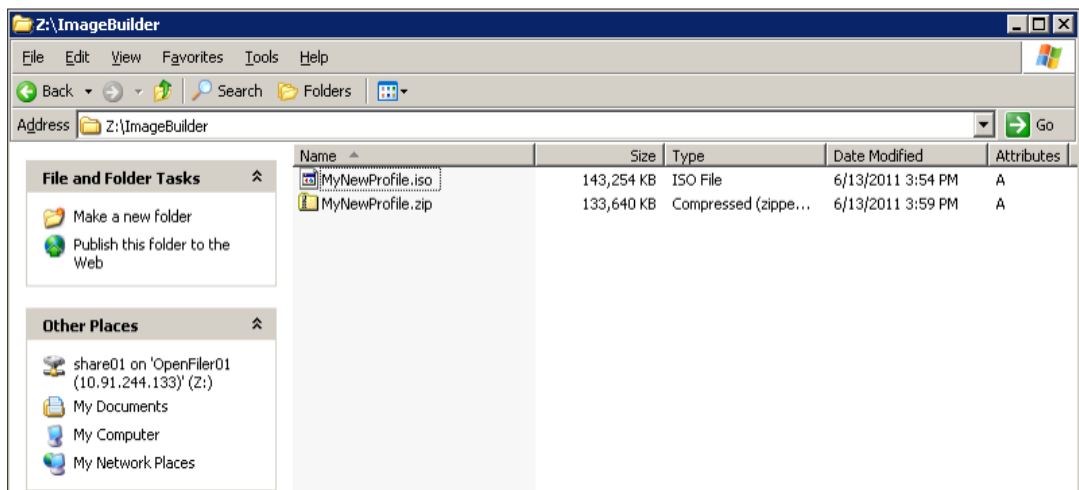


Figure 145. Show Image Profile Exports

Product Documentation

For detailed information regarding installation, configuration, administration, and usage of vSphere Image Builder or other vSphere features, refer to the online documentation: http://www.vmware.com/support/pubs/vs_pubs.html.

Using Storage Performance Statistics

Introduction

vSphere 5.0 introduces several new performance views. These views allow for a quick overview of the current health of your datastores. There are two different types of views: performance and space.

This next section will display how easy datastore monitoring is with vSphere 5.0. There are two basic views as part of the **Datastores and Datastore Clusters** view:

Monitoring Space Utilization of a Datastore

1. Go to the Datastores and Datastore Clusters view.

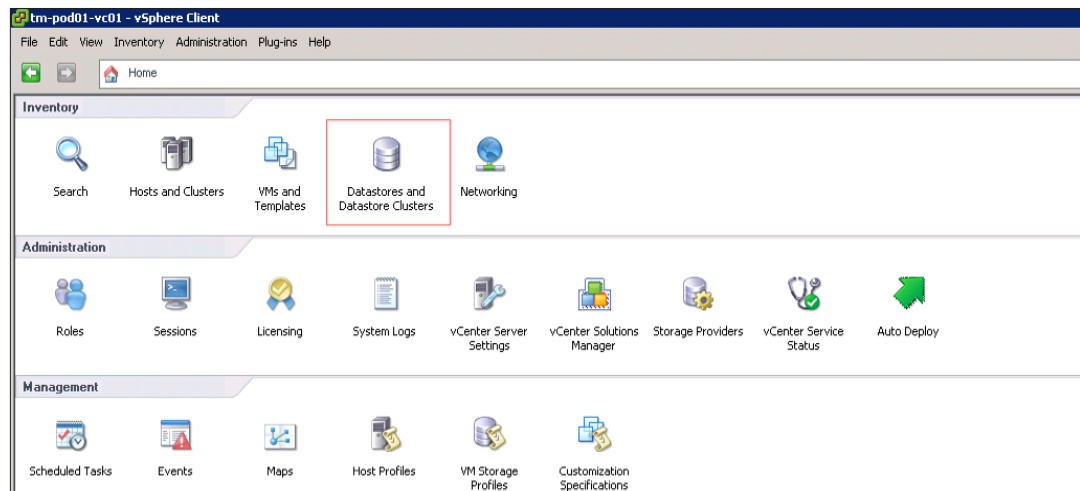


Figure 146.

2. Select a datastore.

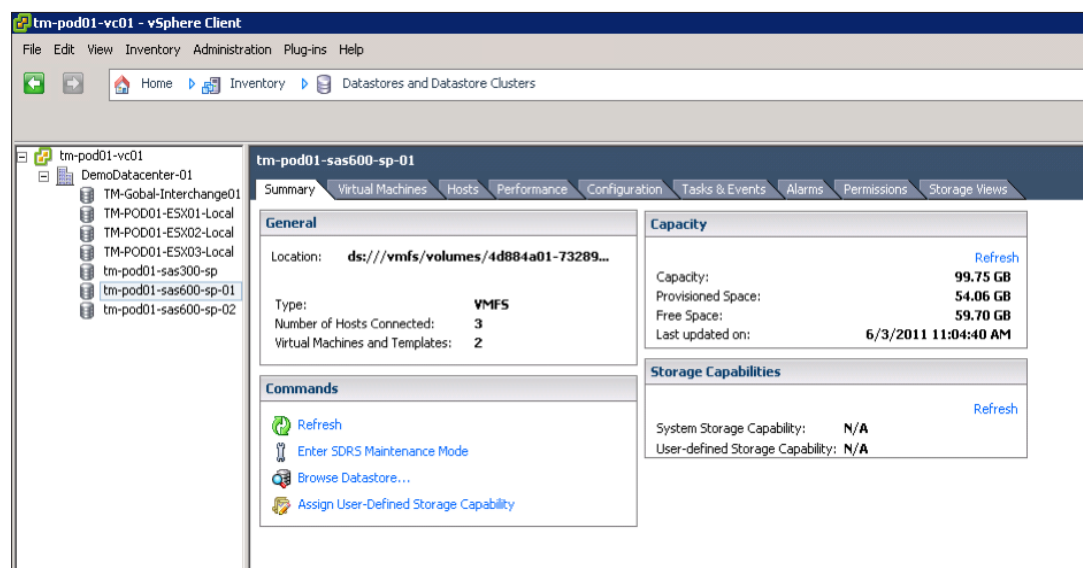


Figure 147.

- Click on the **Performance** tab. This will show you the current Space Utilization statistics for this particular datastore by default.

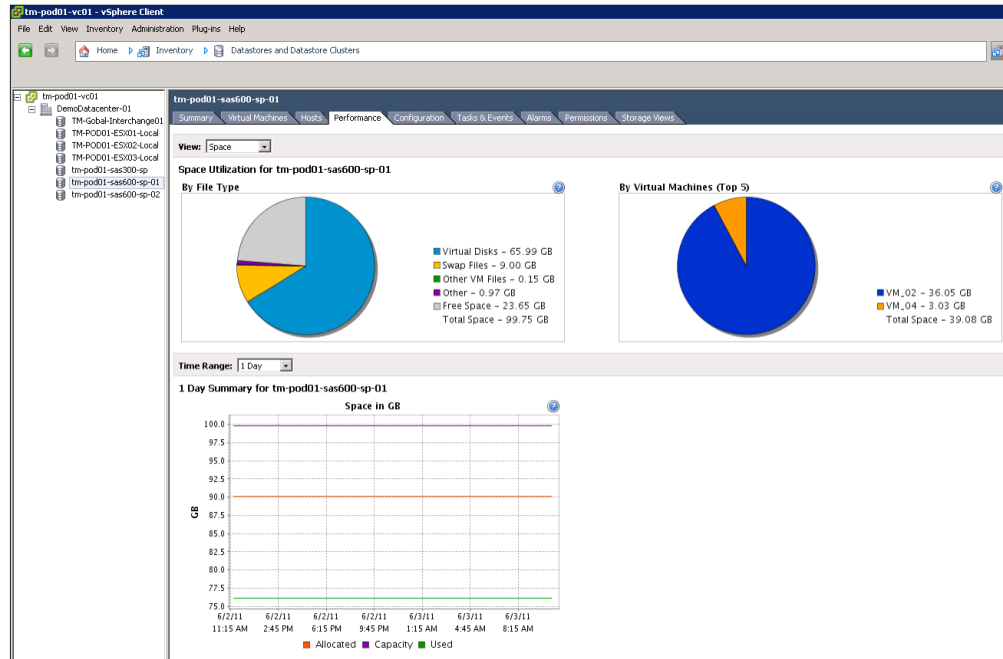


Figure 148.

- Click on **Time Range** to change the range from 1 Day to **1 Week**. This will show if virtual machines have grown or have been migrated to other datastores, and any other trends over the last seven days.

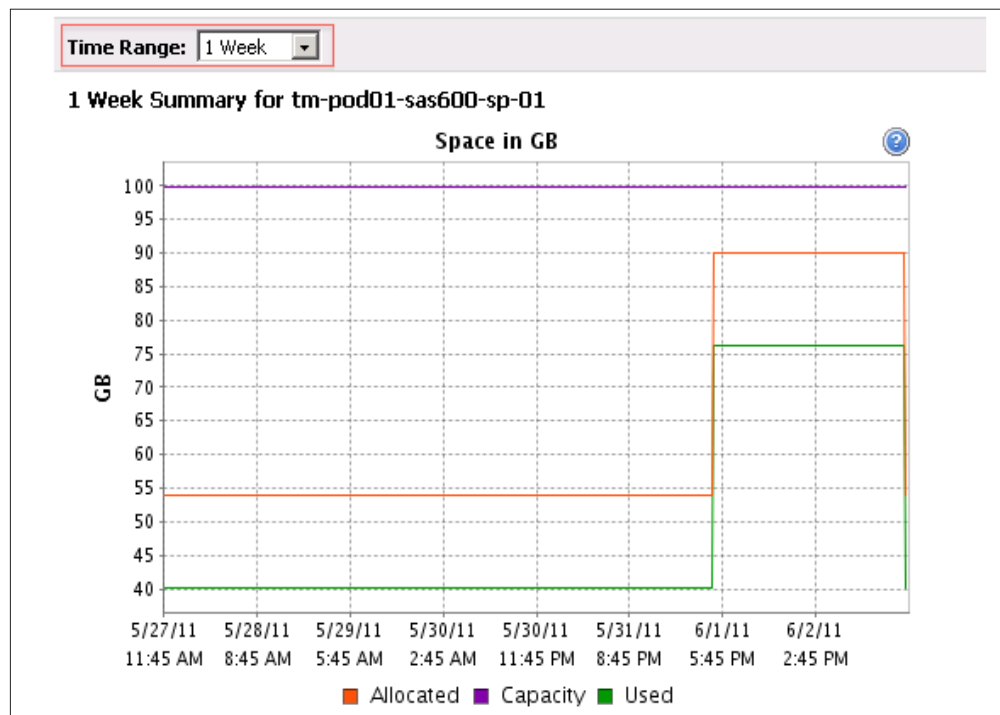


Figure 149.

Monitoring Performance Statistics of a Datastore

The second part of this exercise shows the performance statistics available on the **Datastores and Datastore Clusters** view. These views are showing the most relevant and important metrics to monitor, like **Average Device Latency** **Average Write Latency per Virtual Machine Disk**.

1. Go to the **Datastores and Datastore Clusters** view.

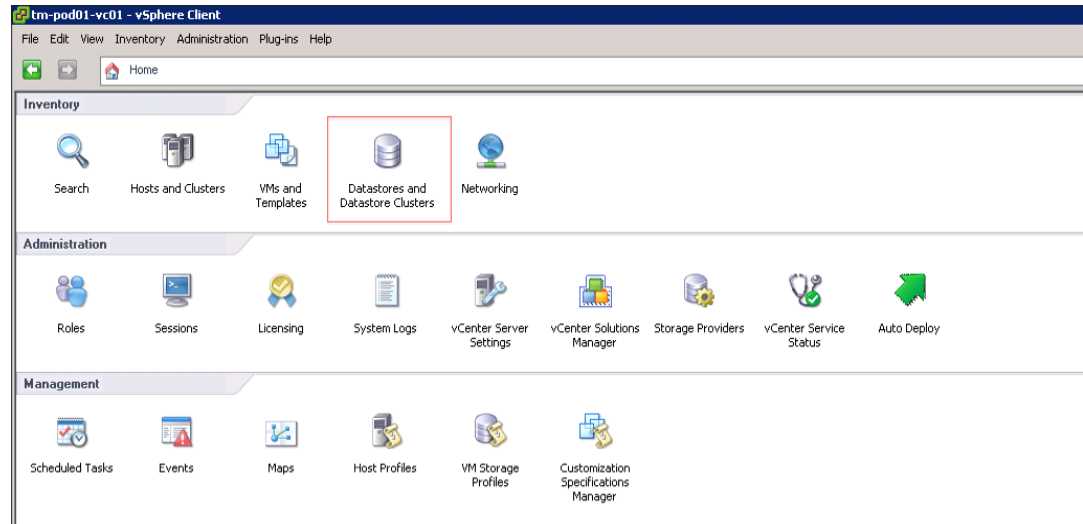


Figure 150.

2. Select a datastore.

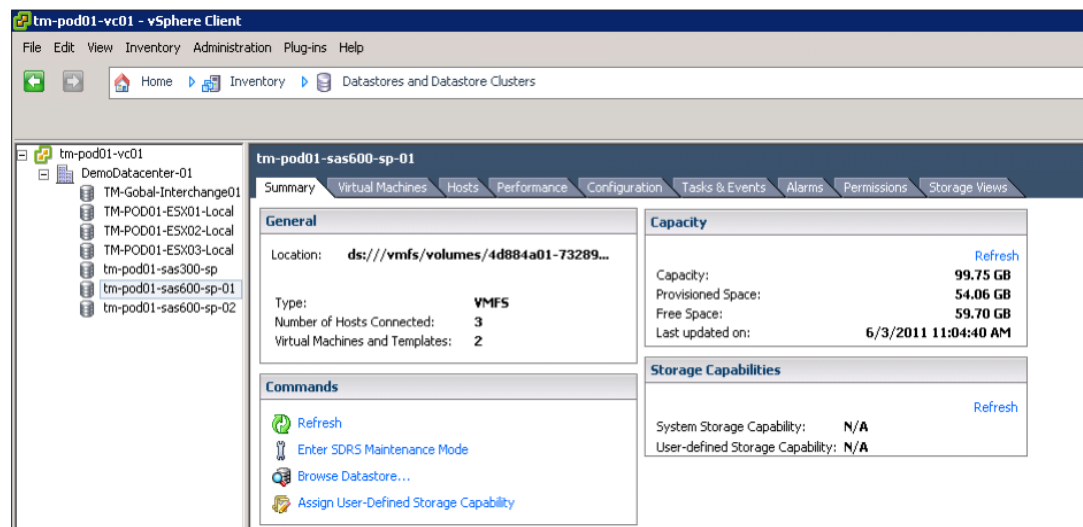


Figure 151.

3. Click on the **Performance** tab and select **Performance** in the **View** drop-down list.

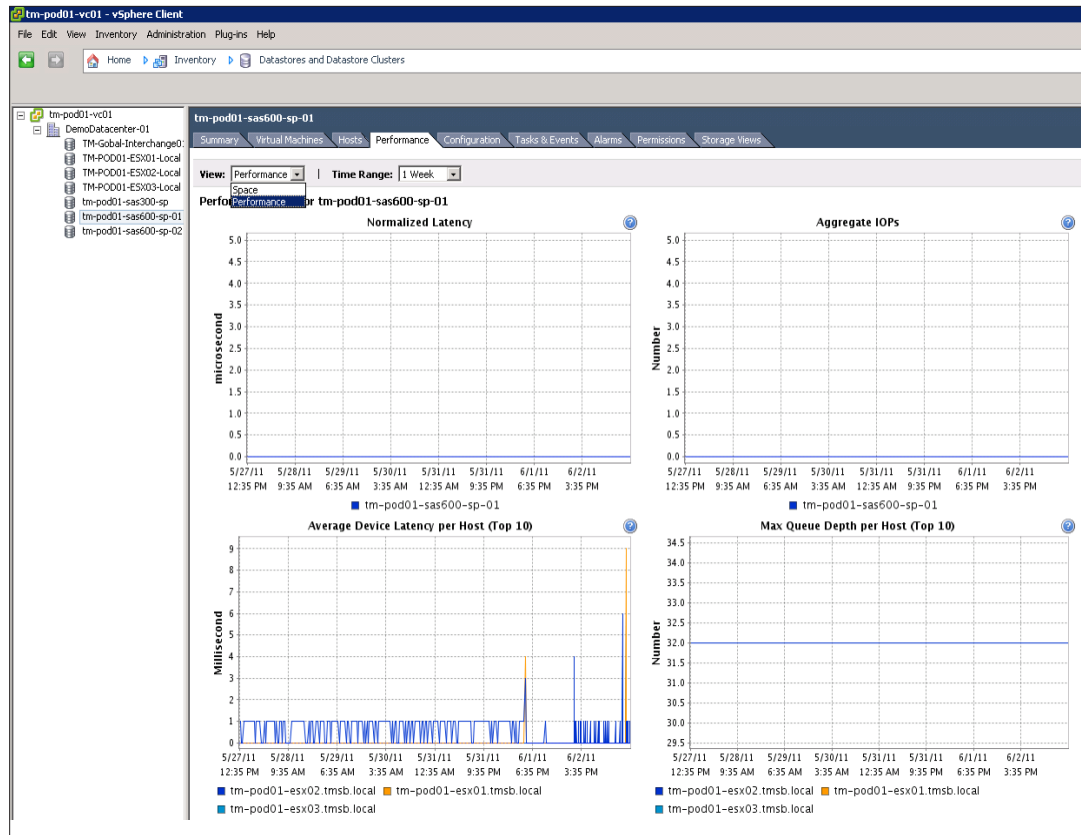


Figure 152.

- You have now successfully completed the Using Storage Performance Statistics exercise.

Help and Support During the Evaluation

This guide provides an overview of the steps required to ensure a successful evaluation of VMware vSphere. It is not meant to be a substitute for product documentation. Refer to the online vSphere product documentation for more detailed information (see the following links). You can also consult the online VMware knowledge base if you have any additional questions. If you require further assistance, contact a VMware sales representative or channel partner.

VMware vSphere and vCenter resources:

- Product documentation:
<http://www.vmware.com/support/pubs/>
- Online support:
<http://www.vmware.com/support/>
- Support offerings:
<http://www.vmware.com/support/services>
- Education services:
<http://mylearn1.vmware.com/mgrreg/index.cfm>
- Support knowledge base:
<http://kb.vmware.com>
- VMware vSphere® PowerCLI Toolkit Community:
http://communities.vmware.com/community/developer/windows_toolkit
(or type Get-VIToolkitCommunity within PowerCLI)
- PowerCLI Blogs:
<http://blogs.vmware.com/vipowershell>

VMware Contact Information

For additional information or to purchase VMware vSphere, the VMware global network of solutions providers is ready to assist. If you would like to contact VMware directly, you can reach a sales representative at 1-877-4VMWARE (650-475-5000 outside North America) or email sales@vmware.com. When emailing, include the state, country and company name from which you are inquiring. You can also visit <http://www.vmware.com/vmwarestore/>.

Providing Feedback

We appreciate your feedback on the material included in this guide. In particular, we would be grateful for any guidance on the following topics:

- How useful was the information in this guide?
- What other specific topics would you like to see covered?
- Overall, how would you rate this guide?

Send your feedback to the following address: tmdocfeedback@vmware.com, with “VMware vSphere 5.0 Evaluation Guide” in the subject line. Thank you for your help in making this guide a valuable resource.



VMware, Inc. 3401 Hillview Avenue Palo Alto CA 94304 USA Tel 877-486-9273 Fax 650-427-5001 www.vmware.com

Copyright © 2011 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. VMware products are covered by one or more patents listed at <http://www.vmware.com/go/patents>. VMware is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies. Item No: VMW-WP-vSPHR-EVAL-GUIDE-VOL1-USLET-101-WEB