WorkshopPLUS Windows PowerShell Desired State Configuration

Module 1: Introduction

Student Lab Manual

Version 1.2

Conditions and Terms of Use

Microsoft Confidential - For Internal Use Only

This training package is proprietary and confidential, and is intended only for uses described in the training materials. Content and software is provided to you under a Non-Disclosure Agreement and cannot be distributed. Copying or disclosing all or any portion of the content and/or software included in such packages is strictly prohibited.

The contents of this package are for informational and training purposes only and are provided "as is" without warranty of any kind, whether express or implied, including but not limited to the implied warranties of merchantability, fitness for a particular purpose, and non-infringement.

Training package content, including URLs and other Internet Web site references, is subject to change without notice. Because Microsoft must respond to changing market conditions, the content should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication. Unless otherwise noted, the companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted herein are fictitious, and no association with any real company, organization, product, domain name, e-mail address, logo, person, place, or event is intended or should be inferred.

Copyright and Trademarks

Contents

Lab 1: Introduction

Introduction

In this lab you will explore the prerequisites for PowerShell Desired State Configuration (DSC) and install the Windows Management Framework (WMF) 5.0 for use throughout the remainder of the course.

Objectives

After completing this lab, you will be able to:

- Define key terminology
- Locate the help topics
- Install WMF 5.0 to get the latest features

Prerequisites

Start all VMs provided for the workshop labs.

Logon to the **PULL** server as **Administrator** with the password **PowerShell5!**

> **NOTE:** If the PULL virtual machine prompts with a license expiration warning, then close the message and run the following command from an elevated PowerShell console:
>
> ```
> slmgr.vbs /rearm
> ```
>
> Ignore the restart requirement, because the server will restart during the first lab.

Estimated time to complete this lab

45 minutes

For more information

The documentation for DSC is available on MSDN:

https://msdn.microsoft.com/en-us/PowerShell/DSC/overview

You will also find the features of DSC described in the release notes:

http://www.microsoft.com/en-us/download/details.aspx?id=48729

A copy of the release notes is available in the “**C:\PShell\WMF 5.0 August 2015 Production Preview**” directory on the PULL server: **Windows Management Framework 5.0 Production Preview Release Notes.docx**

Scenario

As the automation lead for your team, you have been tasked with evaluating PowerShell DSC for use in your organization. You must be able to explain the technology to others and instruct them on how to prepare for the latest features.

NOTE: Use the Windows PowerShell Integrated Scripting Environment (ISE) for typing and editing the PowerShell scripts in this course. The automatic Intellisense will be very helpful in the learning process. If you need to reactivate Intellisense, use the **CTRL SPACE** keyboard shortcut.

Each lab has its own files. For example, the lab exercises for module 1 have a folder at **C:\PShell\Labs\Lab01** on the **PULL** virtual machine. Use these provided files to avoid typing long sections of code.

We recommend that you connect to the virtual machines for these labs through Remote Desktop rather than connecting through the Hyper-V console. When you connect with Remote Desktop, you can copy and paste from one virtual machine to another.

Exercise 1.1: Exploring PowerShell DSC

Objectives

In this exercise, you will:

- Update the PowerShell help files
- Explore the help and documentation
- Define key terms
- Explore implementation considerations

Scenario

You want to view the latest help content as you learn PowerShell DSC.

Prerequisites

The server **PULL** is running a default installation of **Windows Server 2012 R2** with PowerShell v4 and .NET Framework v4.5. PowerShell v5 has the same .NET prerequisite.

Task 1.1.1: Discovering Desired State Configuration Key Terms

1. Turn off **IE Enhanced Security Configuration**. Launch **Server Manager**, then click **Local Server**. On the right side find **IE Enhanced Security Configuration** and click the word **On**. Choose **Off** for both selection, then click **OK**. You would leave this on for production servers, but this is a lab.
2. Open up the **PowerShell Integrated Scripting Environment (ISE)**. Create a new script by pressing **CTRL+N**, or from the menu choose **File** > **New**, or click the NEW icon on the toolbar. It is recommended that you type all of the commands below into a new script, each on a new line.

> **NOTE:** You can highlight a section of code and run it by pressing the **F8** key or execute only the current line simply by pressing the **F8** key (even without highlighting anything, so long as the cursor is on the line).

3. From an elevated PowerShell console run the following command to update the local help files:

```
Update-Help -Verbose
```

> **NOTE:** This requires an internet connection, which is available in the lab on the **PULL** server only.

4. Use the built-in help to discover information about Desired State Configuration. Execute the following command:

```
Get-Help *Desired*
```

Question A: What is the full name of the help topic that was discovered?

____

Question B: What version of Windows PowerShell introduced DSC?

____

5. Read the help topic and answer the questions below:

Question C: **“**DSC provides a set of Windows PowerShell language extensions, new cmdlets, and resources that you can use to ________________ specify how you want the state of your software environment to be configured.”

____

Question D: Identify the two keywords that are used when defining **all** DSC documents:

```
_______________ MyWebConfig
{
     _______________ $MachineName
    {

    }
}
```

____

____

Question E: Write your own definition of the term **idempotent**.

____

____

____

____

Question F: List three of the built-in DSC Resources:

____

____

____

6. Launch the online help:

```
start-process https://msdn.microsoft.com/en-us/PowerShell/DSC/overview
```

Question G: Using the online help, list two **Practical applications** for DSC?

Question H: How are the answers to the two previous questions related?

NOTE: Derive your own answer, as it is not specifically listed in the help topic.

Exercise 1.2: Exploring DSC Cmdlets

Objectives

In this exercise, you will:

- Explore the cmdlets from the **PSDesiredStateConfiguration** module.

Scenario

You already know PowerShell, but you are new to Desired State Configuration. In this exercise you will explore the module and cmdlets available.

Task 1.2.1: Discovering Desired State Configuration Cmdlets

1. Run the following command:

```
Get-Module -Name ps* -ListAvailable
```

Question A: What is the name of the PowerShell Module that we use to manage DSC?

____

2. Review the cmdlets that are part of the **PSDesiredStateConfiguration** module. Run the following command:

```
Get-Command -Module PSDesiredStateConfiguration
```

> **NOTE:** We have not upgraded to PowerShell v5 yet. This is the list of cmdlets in PowerShell v4.

3. Now export the cmdlet list to a text file. We will use this later to compare the new cmdlets after we upgrade to PowerShell v5. Type the command below, using **TAB** completion for the module name and cmdlets:

```
Get-Command -Module PSDesiredStateConfiguration | ForEach-Object Name |
Out-File -FilePath c:\pshell\V4Cmdlets.txt
```

4. Run the following command to review the current **LCM** settings.

```
Get-DscLocalConfigurationManager
```

> **NOTE:** The Local Configuration Manager (LCM) processes configurations.

Question B: Notice a few of the settings. What is the value of **RebootNodeIfNeeded**?

____

5. Run the following command to review the current LCM settings on the Server **MS1**.

> **NOTE:** For more information on CimSession, see **Get-Help about_CIMSession**.

```
Get-DscLocalConfigurationManager -CimSession MS1
```

Question C: What is the **RefreshMode** on the **MS1** server?

____

6. Run the following command to review the built-in DSC resources:

```
Get-DscResource
```

> **NOTE:** We will explore the DSC resources later in Modules 5 and 6.

Exercise 1.3: Upgrading from WMF 4.0 to WMF 5.0

Objectives

In this exercise, you will:

- Upgrade the version of Windows Management Framework (WMF)
- Automate the upgrade using DSC
- Explore the language features of DSC

Scenario

In order to use the full capabilities of Desired State Configuration for our deployments and configurations we should install the highest version of the Windows Management Framework (WMF) that is available. In this exercise we will use preconfigured DSC scripts to upgrade WMF on the Windows servers in our lab environment.

Prerequisites

The prerequisites for WMF 5.0 are .NET Framework 4.5 and a supported operating system. Windows Server 2012 R2 defaults to PowerShell v4 and .NET Framework 4.5. This meets the prerequisite for the upgrade. The WMF 5.0 update package has been provided within the lab files directory.

Task 1.3.1: Running a DSC Configuration to Upgrade WMF

1. Open the **PowerShell ISE** on the **PULL** server.
2. Open the following lab script: **C:\PShell\Labs\Lab01\Lab_01_03_01_Install_WMF_5.ps1**

> **NOTE:** This script will allow you to upgrade from WMF 4.0 to WMF 5.0 on three nodes: PULL, MS1, and MS2. We will cover some of the features seen in this script in upcoming modules. This task provides a real life DSC scenario for discussion.

3. Select and run the region **EXTRACT RESOURCES**. A DSC resource is a special PowerShell module. This region prepares two custom DSC resources that we will use to upgrade the WMF version.
4. Select and run the region **SHARE RESOURCES**. This region creates a share containing the upgrade file and opens the firewall for file traffic over the network.
5. Review the region **PUSH WMF 5 CONFIGURATION**. Notice the **Configuration** and **Node** keywords.

Question D: How many resource types do you see within this configuration?

NOTE: Resources that begin with **x** will be covered later in Module 5.

6. Execute the region **PUSH WMF 5 CONFIGURATION**. Notice that when we call the configuration it creates a directory with the same name as the configuration.

   Question E: What is the output after the configuration has finished?

   ____

   ____

7. Select and run the region **PUSH META CONFIGURATION**. This will apply the LCM configurations to PULL, MS1 and MS2

8. Run the following command to review current LCM settings on node **MS1**:

```
Get-DscLocalConfigurationManager -CimSession MS1
```

   Question F: What is the value of **RebootNodeIfNeeded**? Has this changed from the previous setting?

   ____

9. Select and run the region **PUSH WMF 5 CONFIGURATION TO MS1 AND MS2**. This applies the new configuration and starts the upgrade process to WMF 5.0. The nodes **MS1** and **MS2** will reboot. Notice that since we use the **-Wait** and **-Verbose** switches we can view the process in real time up until the reboot.

10. Select and run the region **PUSH WMF 5 CONFIGURATION TO PULL**. This applies the new configuration and starts the upgrade process to WMF 5.0. The node **PULL** (that you are currently logged on to) will reboot.

11. Log onto **MS2** once it has restarted and review the new PowerShell settings.

12. Launch the PowerShell console.

    Question G: What is the copyright year?

    ____

13. Type **$PSVersionTable**.

    Question H: What version are you now running?

    ____

14. Reconnect to the **PULL** server, launch the PowerShell console, and check the PowerShell versions on all nodes.

```
Invoke-Command -ComputerName ms1,ms2,pull -ScriptBlock {$PSVersionTable.PSVersion}
```

Question I: What version are you now running?

____

____

15. Run the following command in an elevated console to update the local help files after the upgrade:

```
Update-Help -Verbose
```

> **NOTE:** This requires the internet connection available on the **PULL** server.

Question J: Why did we upgrade the versions of WMF from 4.0 to 5.0 in this exercise?

____

____

____

16. Export the cmdlet list from the **PSDesiredStateConfiguration** module to a text file.

```
Get-Command -Module PSDesiredStateConfiguration | ForEach-Object Name |
Out-File -FilePath c:\pshell\V5Cmdlets.txt
```

17. Compare the v4 cmdlets to the v5 cmdlets

```
$V4 = Get-Content -Path c:\pshell\V4Cmdlets.txt

$V5 = Get-Content -Path c:\pshell\V5Cmdlets.txt

Compare-Object -ReferenceObject $v4 -DifferenceObject $v5 -IncludeEqual
```

Question K: What are the new cmdlets?

____

____

> **NOTE:** PowerShell v5 includes many new cmdlets and parameters beyond the PSDesiredStateConfiguration module.

18. Finally, open the following lab script again:
    C:\PShell\Labs\Lab01\Lab_01_03_01_Install_WMF_5.ps1
19. Select and run the last region **CLEAN UP WMF 4.0 DSC**. This removes previous version files to avoid a mismatch between WMF 4.0 and WMF 5.0.