

Running Custom Scripts in the Guest Operating System with Invoke-VMScript

VMware vSphere PowerCLI 5.1 Release 1

In VMware vSphere PowerCLI you can use `Invoke-VMScript` to run a custom script in the guest operating systems of your virtual machines. When running `Invoke-VMScript` does not deliver the expected results, you can determine when and where a problem occurred by knowing how `Invoke-VMScript` works.

To learn more about the features of `Invoke-VMScript`, run `Get-Help Invoke-VMScript`.

Step-By-Step Overview of Running Invoke-VMScript

When you run `Invoke-VMScript`, the cmdlet follows a step-by-step procedure.

- 1 `Invoke-VMScript` detects the `GuestInfo` data object for the specified virtual machine.
- 2 From the `GuestInfo` data object, `Invoke-VMScript` verifies the running status and version of VMware Tools on the specified virtual machine. The cmdlet also determines the guest operating system of the virtual machine.
- 3 If you have not passed a value to the `ScriptType` parameter, `Invoke-VMScript` detects the type of your script automatically.
 - When the guest operating system is Windows, `Invoke-VMScript` runs your script as a PowerShell script.
 - When the guest operating system is Linux, `Invoke-VMScript` runs your script as a BASH script.
- 4 Based on the version of your VMware virtualization software, `Invoke-VMScript` selects a service to run your script with.
 - For vCenter Server 5.0 and later or ESX/ESXi 5.0 and later, `Invoke-VMScript` runs your script through the VIM service.
 - For earlier vCenter Server and ESX/ESXi versions, `Invoke-VMScript` runs your script through the VIX component. The VIX component is included with PowerCLI.

NOTE The VIX component does not support SSPI for guest credentials.

- 5 `Invoke-VMScript` waits for VMware Tools to confirm that the virtual machine is powered on.
- 6 After VMware Tools confirms that the virtual machine is powered on, `Invoke-VMScript` authenticates with the guest operating system. The cmdlet uses the authentication credentials you provided when you called `Invoke-VMScript`.

- 7 `Invoke-VMScript` creates a TEMP file in the virtual machine. The cmdlet uses the TEMP file to store the output of the script.
- 8 Based on the script type, `Invoke-VMScript` modifies the script and redirects the output to the TEMP file.
- 9 Based on the guest operating system, `Invoke-VMScript` selects a script interpreter.
 - When the guest operating system is Windows, the script interpreter is `cmd.exe`.
 - When the guest operating system is Linux, the script interpreter is `/bin/bash`.
- 10 `Invoke-VMScript` runs the script interpreter as a separate process in the guest operating system and passes the modified script to the process.
- 11 When the process completes, `Invoke-VMScript` copies the TEMP file from the virtual machine to the default temporary folder on the Windows system where vSphere PowerCLI is running.
- 12 `Invoke-VMScript` reads the contents of the local copy of the TEMP file and stores them in a variable.
- 13 `Invoke-VMScript` deletes the copies of the TEMP file from the local and the virtual machine.
- 14 `Invoke-VMScript` returns the output of the script.

Running a PowerShell Script with Invoke-VMScript

When your script is a PowerShell script, `Invoke-VMScript` uses `cmd.exe` to start PowerShell. `Invoke-VMScript` modifies the PowerShell script to escape the special characters that the script contains. In the script interpreter, the cmdlet runs `powershell.exe -Command`. `Invoke-VMScript` passes the modified script to the `Command` parameter.

To redirect the script output, `Invoke-VMScript` encodes the `powershell.exe -Command` command and passes it to another PowerShell process.

To modify your PowerShell script, `Invoke-VMScript` escapes the special characters that the script contains in a particular order.

- 1 A single quotation mark (') is replaced with a double quotation mark (").
- 2 A string of a backslash and a double quotation mark (\") is replaced with a string of two backslashes and a double quotation mark (\\").
- 3 A double quotation mark (") is replaced with a string of a backslash and a double quotation mark (\").

Running a BAT Script with Invoke-VMScript

When your script is a BAT script, `Invoke-VMScript` uses `cmd.exe` to run the script. `Invoke-VMScript` modifies the BAT script to escape the special characters that the script contains. The cmdlet surrounds the modified script with double quotation marks (") and passes it to the script interpreter.

To modify your BAT script, `Invoke-VMScript` escapes only the special characters that indicate a new line. A string for a new line (`\r\n` or `\n`) is replaced with a string of a space, an ampersand, and a space (&).

Running a BASH Script with Invoke-VMScript

When your script is a BASH script, `Invoke-VMScript` uses `/bin/bash` to run the script. `Invoke-VMScript` modifies the BASH script to escape the special characters that the script contains. The cmdlet surrounds the modified script with strings of a backslash and a double quotation mark (`\`) and passes it to the script interpreter.

To modify your BASH script, `Invoke-VMScript` escapes the special characters that the script contains in a particular order.

- 1 A string for a new line (`\r\n`) is replaced with a string of a space, a semicolon, and a space (`;`).
- 2 A string of a backslash and a double quotation mark (`\`) is replaced with a string of a single quotation mark, a double quotation mark, and a single quotation mark (`"`).
- 3 A double quotation mark (`"`) is replaced with a string of three backslashes and a double quotation mark (`\\\"`).

If you have comments about this documentation, submit your feedback to: docfeedback@vmware.com

VMware, Inc. 3401 Hillview Ave., Palo Alto, CA 94304 www.vmware.com

Copyright © 2009–2012 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: EN-000909-00
