

Requisitos previos de ejecución de un script

Entre las novedades de PowerShell v2, también podemos contar con los requisitos previos de ejecución. En efecto, a partir de la versión 2 de PowerShell es posible impedir la ejecución de un script si las condiciones no se cumplen. Dichas condiciones podrán referirse a los ámbitos siguientes :

Tipo de filtros	Sintaxis
La versión de PowerShell utilizada, v2 o posterior.	<code>#requires -Version <número de versión></code>
La versión de un snap-in (componente conectable).	<code>#requires -PsSnapIn <nombre del snapin> [-Version <número de versión>]</code>
El interprete de comandos.	<code>#requires -ShellId <Id del shell></code>

Para indicar estos requisitos previos, nada más sencillo que colocar en el inicio del script, o bien de una línea, el símbolo almohadilla y la palabra clave «requires» seguido por los parámetros deseados, ver tabla superior. Tomemos el caso sencillo de un script que debe utilizar el snap-in VMware entregado con vSphere PowerCLI que permite administrar un conjunto de servidores VMware ESX a partir de PowerShell.

```
#requires -PsSnapIn vmware.VimAutomation.Core
```

Si el snap-in no está correctamente cargado, entonces aparece el mensaje siguiente:

```
PS > .\MiScript.ps1
No se puede ejecutar el script 'MiScript.ps1' porque faltan los siguientes
complementos que están especificados por las instrucciones
"#requires" del script: vmware.VimAutomation.Core.
En línea: 1 Carácter: 15
+ .\MiScript.ps1 <<<<
    + CategoryInfo          : ResourceUnavailable: (MiScript.ps1:String) [],
ScriptRequiresException
    + FullyQualifiedErrorId : ScriptRequiresMissingPSSnapIns
```

Al contrario, es decir, si el snap-in está cargado antes de la ejecución del script, entonces este último no encontrará un enlace con la ejecución.

```
PS > Add-PSSnapin vmware.VimAutomation.Core
PS > .\miScript.ps1
Inicio del script...
```

Segundo ejemplo, este de la versión del interprete de comandos. Tomemos un caso muy simple de un script que debe utilizar el shell PowerShell y no el correspondiente a Exchange 2007 (Exchange Shell). El requisito previo se indica entonces de la siguiente manera:

```
#requires -ShellId Microsoft.PowerShell
```

Si el requisito previo no se cumple en la ejecución del script entonces se producirá un mensaje de error impidiendo la ejecución.



Truco: El ShellID de la consola PowerShell por defecto es «Microsoft.PowerShell». De una forma general, el ShellID puede obtenerse vía la visualización del contenido de la variable \$ShellID.

Respecto al caso de la versión de PowerShell, los más atentos de ustedes habrán observado que hasta el momento esta

restricción ha tenido poco interés. En efecto, las restricciones comienzan con el símbolo almohadilla, lo que indica que la versión 1.0 que interpreta la almohadilla como una línea de comentario, no está en condiciones de determinar que se trata de un filtro en la versión de PowerShell. Esta funcionalidad de filtrado de la versión será muy útil para las futuras versiones.



La utilización de la instrucción Requires puede acumularse varias veces en un mismo script, sin embargo puede utilizarse únicamente en los scripts.
