

Capítulo 10: La recuperación del sitio, sin VMware SRM



JmGVirtualConsulting.com

virtually anything is possible

<http://www.JmGVirtualconsulting.com>



El Blog de Virtualización en Español

Con Jose Maria Gonzalez, VCP, vExpert, RHCE, MCSE, MCSA, CPSA

<http://www.josemariagonzalez.es>

Introducción

Una de las ironías o paradojas interesantes que descubrí a la hora de escribir este libro, fue ¿Que pasa si en el momento de ejecutar mi plan DR, VMware Recovery Manager falla o no está disponible?. Dicho de otro modo, ¿cuál es nuestro plan de recuperación para SRM!?. Bromas aparte, se trata de una pregunta que merece la pena considerar. No tiene mucho sentido el uso de cualquier tecnología sin un plan B, por si el plan A no funciona como esperábamos.

Dado que la clave de cualquier plan de recuperación es la replicación de datos a un sitio de recuperación, el elemento más importante está a cargo de su cabina de almacenamiento y no por VMware SRM. Recuerde que todo lo que VMware SRM está haciendo es automatizar un proceso manual. Por lo que detrás de este capítulo, hay realmente dos grandes agendas. La primera es, como hacer manualmente todo lo que hace VMware SRM en el caso de que nuestro plan A no funcione. Y la segunda es, mostrarle que SRM es increíblemente útil para automatizar este proceso. Espero que pueda ver en este capítulo, lo difícil que es la vida sin VMware Site Recovery Manager. Como cualquier proceso de automatización o de secuencias de comandos, usted no ve realmente las ventajas hasta que sepa como es el proceso manual.

Con esto en mente, podría haber empezado con este capítulo en el capítulo 1 o 2, pero pensé que usted desearía profundizar en SRM, que es el tema de este libro, y guardar este contenido para el final. Esto también le dará una idea de lo que SRM hace en el fondo, que es hacer su vida mucho más fácil. La gran ventaja de SRM, para mí, es que crece y reacciona a los cambios de su sitio protegido, algo a lo que un proceso manual tendría niveles mucho más altos de mantenimiento para lograrlo.

Como parte de la preparación de este capítulo, he decidió suprimir la protección de los grupos y los planes de recuperación asociados a nuestra configuración bi-direccional de mi sitio de recuperación (Reading).

La recuperación manual de máquinas virtuales exigirá alguna gestión de almacenamiento, como detener el actual ciclo de replicación y promover el "volumen remoto" en un volumen primario o principal de lectura-escritura. Mientras que SRM hace esto automáticamente por usted, a través del SRA de su proveedor de almacenamiento, en una recuperación manual, tendrá que hacerlo por sí mismo. Esto es asumiendo que todavía tienen acceso a la cabina en el sitio de protección, como ocurre con una ejecución planeada de su plan DR. Además, una vez que la replicación se ha detenido, tendremos que conceder acceso a los servidores ESX del sitio de recuperación a la última instantánea buena que fue tomada.

Por parte de los servidores ESX, una vez concedido el acceso a los volúmenes, estos tendrá que ser manualmente re-escaneados para asegurarse de que el volumen VMFS ha sido montando en los servidores ESX. Basándose en nuestras necesidades y la visibilidad de la LUN, tendremos la opción de no hacer el *resignature* o forzar una *resignature* del volumen VMFS.

Después de haber "lidiado" con la parte de almacenamiento, tendremos que editar el archivo VMX de cada máquina virtual y mapear este a la red correcta. Después de hacer esto, estaríamos en condiciones de empezar a añadir cada máquina virtual en el sitio de recuperación y por cada máquina tenemos que decirle al cliente Vi que grupo, carpeta y *resource pool* va utilizar.

En un mundo ideal, parte de la gestión de la máquina virtual podría ser hecha vía secuencia de comandos, utilizando diferentes kits de desarrollo de software de VMware como Perl Scripting ToolKit, el PowerShell Scripting Toolkit o el SDK de vCenter con el



JmGVirtualConsulting.com

virtually anything is possible

<http://www.JmGVirtualconsulting.com>



El Blog de Virtualización en Español

Con Jose Maria Gonzalez, VCP, vExpert, RHCE, MCSE, MCSA, CPSA

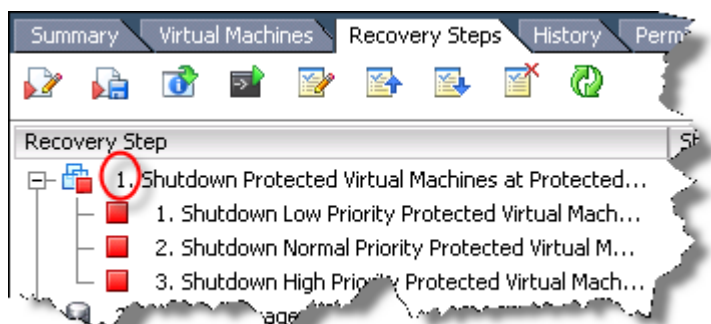
<http://www.josemariagonzalez.es>

lenguaje de su elección - VB, C#, etc. Tengo la intención de utilizar el PowerShell Toolkit para VMware como un ejemplo.

Como podrá ver, las secuencias de comandos, es un proceso muy laborioso y tedioso. Fundamentalmente, es un proceso muy lento por lo que impactara sobre la rapidez en su proceso de recuperación. Piense en todas las RTOs y RPOs.

Para una recuperación no planificada

Para la recuperación no planificada, hay que apagar las máquinas virtuales que se están ejecutando en producción, antes de administrar el almacenamiento. Si recuerda, los primeros pasos de cualquier recuperación prevista, es apagar las máquinas virtuales en el sitio protegido.



Si usted está haciendo un *failover* manual con fines de prueba no es necesario.

Gestionar el almacenamiento

Sin un SRA, tendremos que participar más con las herramientas del vendedor de almacenamiento para el control de las instantáneas y la replicación. Este área es muy específica de los proveedores, por lo que le aconsejo se lea la documentación. En el caso de VSA Lefthand Networks, serían los siguientes pasos. Actualmente, el grupo RecoveryManagementGroup (Reading) tiene la copia primera y el grupo ProtectedManagement (Londres) tiene la copia remota, mantenido por un ciclo de instantáneas programado. Necesitaré dar acceso a los servidores ESX a la última instantánea o volumen replicado, añadiendo esta a una lista de volumen existente.

- 1. Entre en el VSA Lefthand Networks como administrador**
- 2. Seleccione el ProtectedManagementGroup y, a continuación, seleccione la pestaña Volume Lists**
- 3. Seleccione un volumen existente de la lista y añada la última instantánea a la lista**



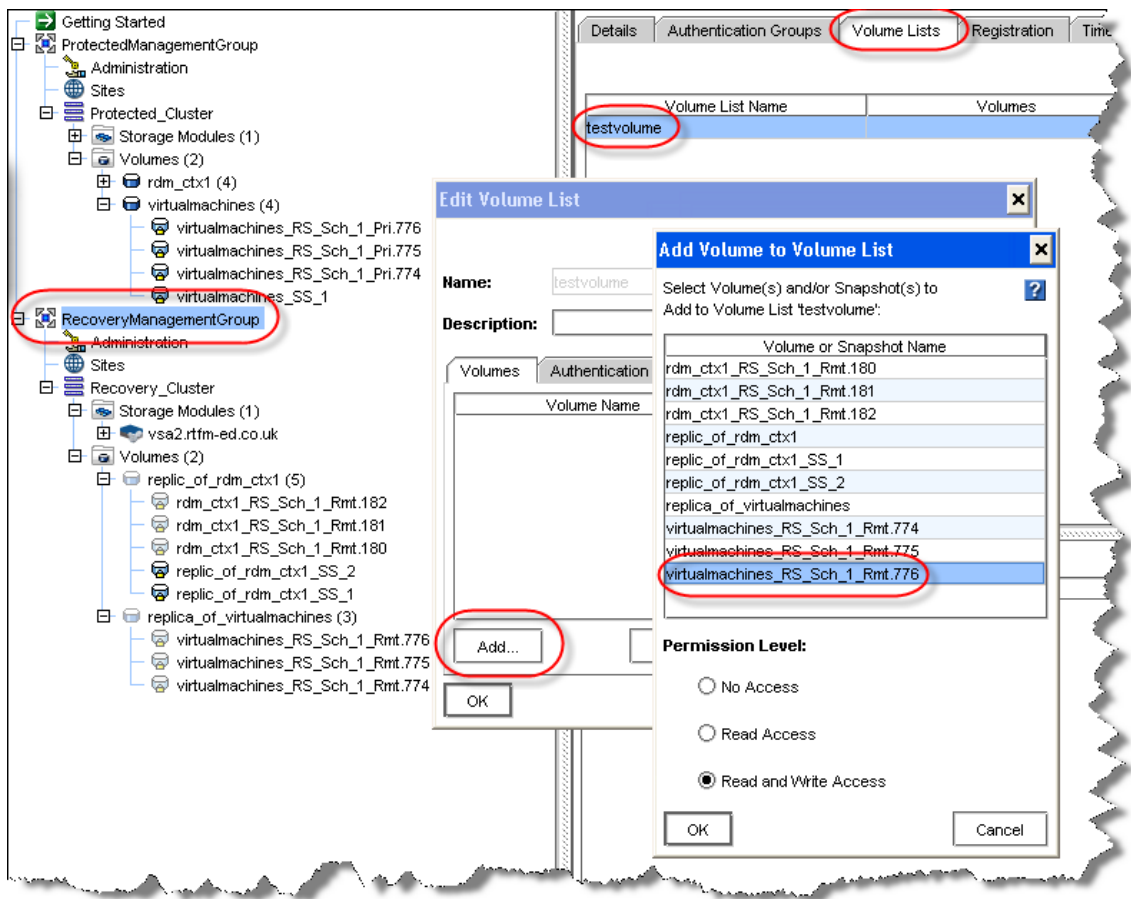
JmGVirtualConsulting.com
virtually anything is possible

<http://www.JmGVirtualconsulting.com>



El Blog de Virtualización en Español
Con Jose Maria Gonzalez, VCP, vExpert, RHCE, MCSE, MCSA, CPSA

<http://www.josemariagonzalez.es>



Nota:

En el ejemplo anterior estoy dando acceso a mi servidor ESX en Londres a la última instantánea de mi volumen virtualmachines (776).

VMware PowerShell Toolkit

Las siguientes secciones discuten el proceso manual de añadir las máquinas virtuales en vCenter y dejarlas listas para encenderlas. También he decidido mostrarle cómo hacer las mismas tareas con PowerShell.

Lo que sigue es casi una guía de configuración para PowerShell. En primer lugar, descargue e instale, Windows PowerShell V2 Community Technology Preview 2 (CTP2). Pondría la URL en el libro, pero es demasiado larga y, en última instancia, es muy probable que cambie. En la actualidad, la versión oficial de Microsoft PowerShell es la versión 1. Habrá una versión 2 muy pronto aunque en la actualidad sólo está en fase de revisión.

Después, usted tendrá que descargar e instalar el VMware PowerShell Toolkit (en realidad se llama Vi3 Toolkit):

<http://www.vmware.com/support/developer/windowstoolkit/beta/windowstoolkit-200803-releasenotes.html>

Luego usted tendrá que abrir sesión de PowerShell, y descargar VMware PowerShell Community Extensions.



JmGVirtualConsulting.com
virtually anything is possible

<http://www.JmGVirtualconsulting.com>



El Blog de Virtualización en Español
Con Jose Maria Gonzalez, VCP, vExpert, RHCE, MCSE, MCSA, CPSA

<http://www.josemariagonzalez.es>

```
(new-object net.webclient).DownloadString("http://communities.vmware.com/servlet/JiveServlet/downloadBody/6051-102-1-3481/Extensions.psm1") > $env:temp/Extensions.psm1
add-module $env:temp/Extensions.psm1
```

Una vez descargado, usted podrá instalarlo con el siguiente comando:

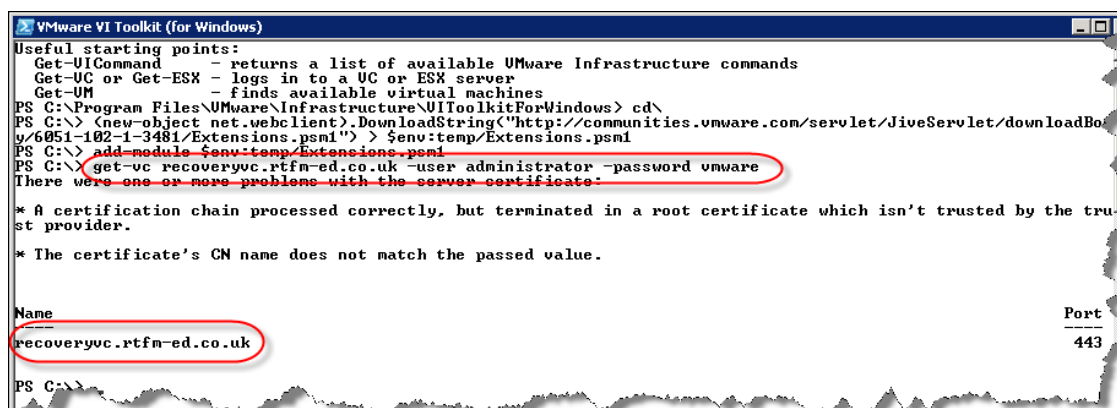
```
add-module $env:temp/Extensions.psm1
```

PRECAUCIÓN:

Descubrí que tenía que repetir este proceso cada vez que abría una nueva sesión de PowerShell.

Para conectarse y acceder al vCenter, use el comando get-vc de este modo:

```
get-vc londonvc.rtfm-ed.co.uk -user administrator -password vmware
```



```
VMware VI Toolkit (for Windows)
Useful starting points:
Get-UICommand - returns a list of available VMware Infrastructure commands
Get-UC or Get-ESX - logs in to a UC or ESX server
Get-UM - finds available virtual machines
PS C:\Program Files\VMware\Infrastructure\UIToolkitForWindows> cd\
PS C:\> (new-object net.webclient).DownloadString("http://communities.vmware.com/servlet/JiveServlet/downloadBody/6051-102-1-3481/Extensions.psm1") > $env:temp/Extensions.psm1
PS C:\> add-module $env:temp/Extensions.psm1
PS C:\> get-vc londonvc.rtfm-ed.co.uk -user administrator -password vmware
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
-----
recoveryvc.rtfm-ed.co.uk                443

PS C:\>
```

ADVERTENCIA:

Me imagino que esto cambie antes de que el libro sea publicado. Si usted tiene dificultades para encontrar los diferentes programas y enlaces, este es un buen lugar para empezar. Yo he usado esta web como punto de partida para descargar todos los archivos binarios que necesitaba.

<http://blogs.vmware.com/vipowershell/2008/06/fun-with-powers.html>

SUGERENCIA:

Entiendo lo difícil que es escribir todos estos códigos a mano, y siendo este un libro en papel, no hay opciones de cortar y pegar.

Por lo tanto, he cogido todos estos ejemplos de PowerShell y los he puesto en un archivo de texto. Puede descargar este archivo de texto desde el sitio web para poder cortar y pegar, y cambiar las variables tales como los nombres de sus máquinas virtuales y recursos piscina nombres.

<http://www.josemariagonzalez.es/srm.html>

Escanear las HBAs de cada servidor ESX

Usted debe de saber, más que suficiente, cómo re-escanear un servidor ESX, bien desde la GUI o mediante el CLI. Lo que usted necesita es que aparezca un nuevo volumen VMFS.



JmGVirtualConsulting.com

virtually anything is possible

<http://www.JmGVirtualconsulting.com>



El Blog de Virtualización en Español

Con Jose Maria Gonzalez, VCP, vExpert, RHCE, MCSE, MCSA, CPSA

<http://www.josemariagonzalez.es>

Storage				Refresh	Remove	Add Storage
Identification	Device	Capacity				
esx3:storage1	vmhba0:0:0:9	52.75 GB	64			
isos	vmhba1:1:0:1	67.50 GB	17			
sanstorage	vmhba1:0:0:1	101.25 GB	9			
snap-398cdef9-london_virtualmachines	vmhba32:50:0:1	1,023.75 GB	99			
templates	vmhba1:3:0:1	135.00 GB				

Este re-escaneo tiene que ser hecho una vez por cada servidor ESX, y sería muy laborioso tener que hacerlo a través del cliente Vi. Por supuesto, usted podría entrar con PuTTY y usar el comando `esxcfg-rescan`. Personalmente, prefiero utilizar el *Remote CLI* para Windows, el cual puede ser instalado en el mismo sistema de gestión donde está instalado el PowerShell de VMware. Con el RCLI para Windows podemos utilizar el script `esxcfg-rescan.pl`

`esxcfg-rescan.pl --server esx1 --username root --password vmware vmhba32`

El siguiente comando de PowerShell, rescanea todos los hosts ESX en vCenter, lo cual es mucho más eficiente desde una perspectiva de secuencias de comandos.

`get-vmhost | vmhoststorage-get-rescanallhba`

```
PS C:\> get-vmhost | get-vmhoststorage -rescanallhba
Softw FileSystemVolumeInfo          ScsiLun
areIS
csiEn
abled
-----
True  {sanstorage, isos, snap-398cdef9... {key-vm.host.ScsiDisk-vmhba1:3:0, ke...
True  {esx3:storage1, sanstorage, snap... {key-vm.host.ScsiDisk-vmhba1:3:0, ke...
```

Nota:

La sintaxis del comando de PowerShell anterior es relativamente fácil de explicar. `Get-vmhost` recupera todos los nombres de los servidores ESX en el vCenter y esto es enviado al comando `get-vmhoststorage` para re-escanear todos los servidores ESX. `Get-vmhoststorage` soporta la opción `-rescanALLhba` que hace exactamente lo que usted piensa.

Usted podría encontrar que el volumen VMFS no refleja el nombre original del DataStore. Esto depende en gran medida de si se ha hecho el *resigned*, ya sea mediante SRM o manualmente utilizando la configuración avanzada del servidor ESX. Si usted desea cambiar el nombre del volumen VMFS como lo hace SRM, después de una *resignature*, puede utilizar este comando de PowerShell.

`set-datastore -datastore (get-datastore *london_virtualmachines) -name london_virtualmachines`

```
PS C:\> set-datastore -datastore (get-datastore *london_virtualmachines) -name london_virtualmachines
Name                FreeSpaceMB    CapacityMB
-----
london_virtualmachines 1015751        1048320
```

Nota:

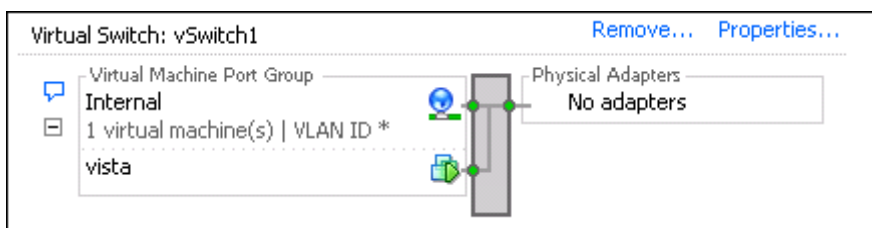
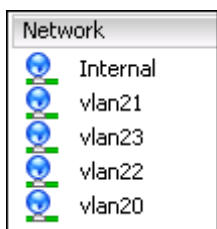
`Set-datastore` se puede utilizar para cambiar el nombre del volumen VMFS y el nombre del `dataStore` con la opción `-name`. Usado en conjunción con el comando `get-datastore`



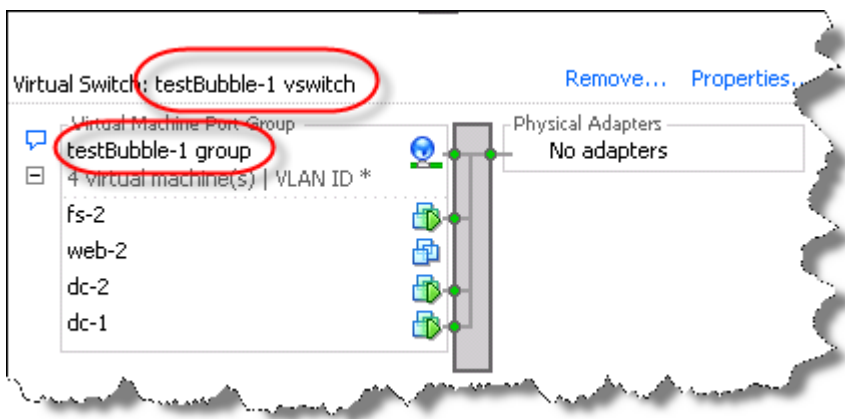
cmdlet, podemos buscar (usando un comodín *) por el volumen VMFS que incluya la cadena "london_virtualmachines", y cambiarle el nombre original al volumen VMFS y datastore.

Crear una red interna para las pruebas

Es parte de mi configuración estándar crear en todos mis servidores ESX, un port group llamado "internal", el cual es un switch dedicado que no tiene mapeada ninguna tarjeta física.



Sin embargo, usted podría querer emular la forma en la que SRM hace sus pruebas para los planes de recuperación, creando una red de test llamada "testbubble".



La creación de los switches virtuales en la versión beta de VMware PowerShell es bastante complicado. Este hecho fue resaltado por el equipo de VMware PowerShell, y es probable que sea más fácil en la próxima versión. El comando PowerShell para crear esto es un poco largo:

```
{noformat}
$MyvSwitchName = "testBubble-1 vswitch"
$MynumPorts = 64
$MyPortGroupName = "testBubble-1 group"
$MyvlanID = 0
```

```
Get-VMHost | %{Get-View (Get-View $_.ID).configmanager.networkSystem} | %{
    $vSwitchSpec = New-Object vmware.vim.HostVirtualSwitchSpec
```



JmGVirtualConsulting.com
virtually anything is possible

<http://www.JmGVirtualconsulting.com>



El Blog de Virtualización en Español
Con Jose Maria Gonzalez, VCP, vExpert, RHCE, MCSE, MCSA, CPSA

<http://www.josemariagonzalez.es>

```

$vSwitchSpec.numPorts = $MynumPorts
$_.AddVirtualSwitch($MyvSwitchName,$vSwitchSpec)

$PortgroupSpec = New-Object vmware.vim.hostportgroupspec
$PortgroupSpec.vswitchname = $MyvSwitchName
$PortgroupSpec.Name = $MyPortGroupName
$PortgroupSpec.vlanID = $MyvlanID
$PortgroupSpec.policy = New-Object VMware.Vim.HostNetworkPolicy
$_.AddPortGroup($PortgroupSpec)
}
{noformat}

```

Nota:

En este caso, en la parte superior de la secuencia de comandos de PowerShell, tenemos un número de variables que se definen como el nombre del switch virtual, número de puertos (16,32,64 hasta 1024) y, el primer nombre portgroup sin ajustes VLAN. Este script no crea ningún switch virtual. Para correr este script, usted tiene que abrir una sesión de PowerShell y guardar el contenido en un archivo de comandos PowerShell (.Ps1) con:

notepad vswitch.ps1

y para ejecutar el script debería escribir:

. \ vswitch.ps1

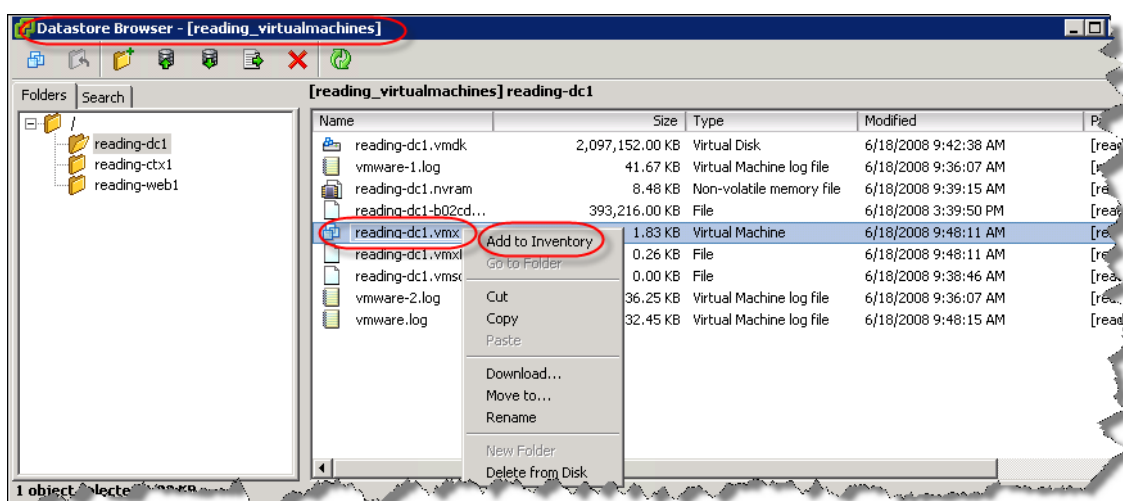
```

PS C:\> notepad vswitch.ps1
PS C:\> .\vswitch.ps1
noformat
noformat
PS C:\>

```

Añadir máquinas virtuales en el Inventario

1. En uno de los servidores ESX
2. **Navegue por datastore** que contiene las máquinas virtuales
3. Haga **clic derecho en el archivo VMX** y elija **Add to Inventory**



JmGVirtualConsulting.com
virtually anything is possible

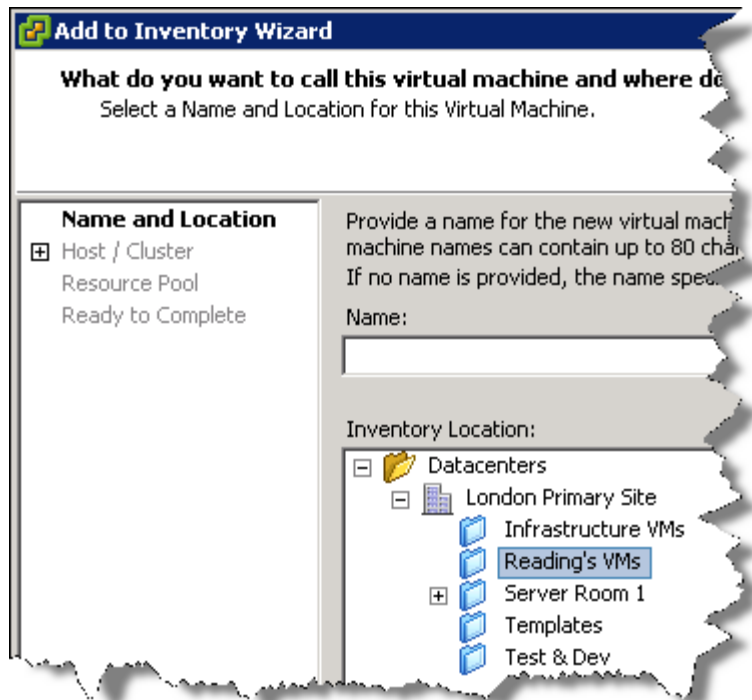
<http://www.JmGVirtualconsulting.com>



El Blog de Virtualización en Español
Con Jose Maria Gonzalez, VCP, vExpert, RHCE, MCSE, MCSA, CPSSA

<http://www.josemariagonzalez.es>

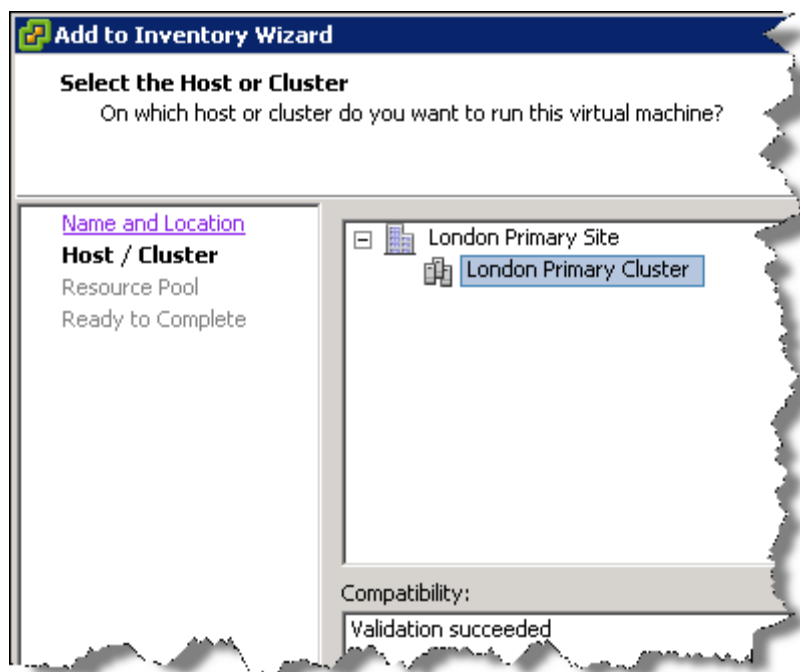
4. En el subsiguiente cuadro de diálogo, seleccione un DataCenter y una carpeta en donde almacenar su máquina virtual



Nota:

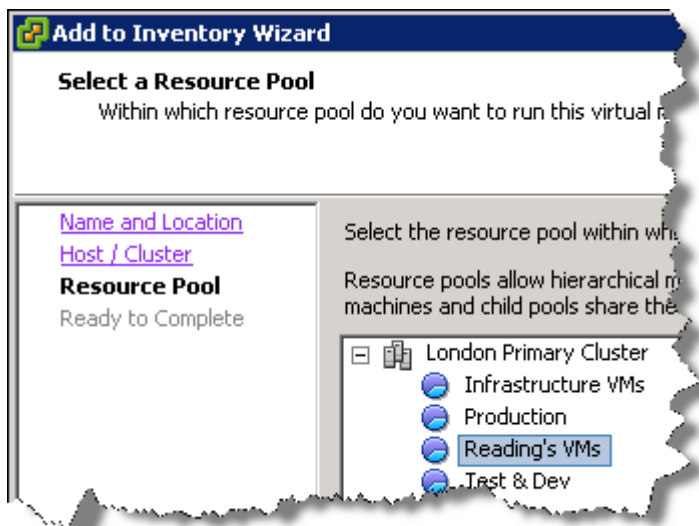
Usted no tiene que especificar un nombre para su máquina virtual, lo que provoca que vCenter lea el archivo VMX por el campo displayName = "..."

5. Seleccione el Servidor ESX o Cluster



6. A continuación, **seleccione un resource pool** para la máquina virtual





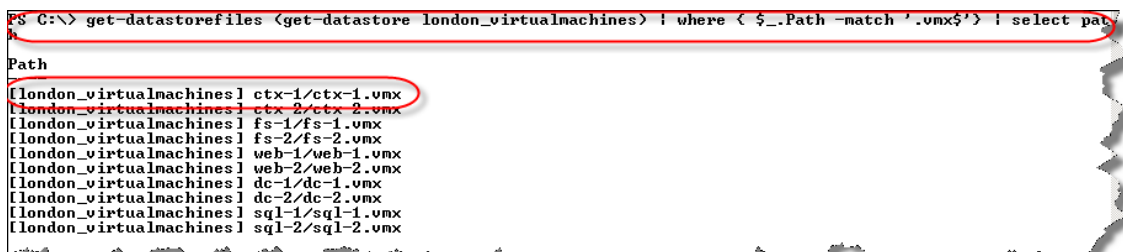
Nota:

Usted debería ser capaz de encender la máquina virtual. Usted tendrá que cambiar manualmente su dirección IP en el sistema operativo invitado. Es posible automatizar la adición de una máquina virtual al servidor ESX (no a un clúster), utilizando la línea de comandos del servidor ESX llamado vmware-cmd. Lamentablemente, este comando no puede manejar los metadatos de vCenter, tales como la ubicación de la carpeta y los *resource pools*.

Recuerde que usted tendrá que repetir estos pasos por cada máquina virtual que debe ser recuperada.

Quizás, una mejor forma de hacerlo es utilizando algunos PowerShell. Podemos usar el cmdlet `get-datastorefile` y `get-datastore` para proporcionar una lista de todas las máquinas virtuales en el DataStore.

get-datastorefiles (get-datastore london_virtualmachines) | where { \$_.Path -match '.vmx\$' } | select path



Nota:

Este fragmento de PowerShell, listas todas las carpetas y archivos (`get-datastorefiles`) en un determinado volumen VMFS, el cual es recuperado por el `get-datastore` cmdlet. Esto es filtrado a través de una búsqueda (seleccione ruta) para mostrar únicamente la información sobre la ruta, donde la carpeta contiene un archivo VMX. Por último, la salida es filtrada de nuevo, sólo para mostrar rutas de archivos en este formato.

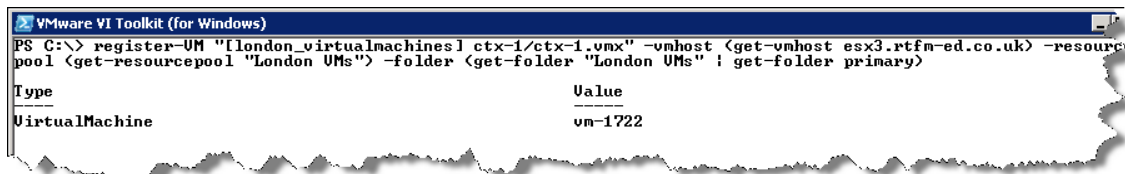
[london_virtualmachines] / ctx1/ctx1.vmx

Una vez que tenemos el camino, podemos pensar en tratar de registrar la máquina virtual. Hay un `register-VM` cmdlet, que podemos utilizar para manejar todo el



proceso de registro, incluyendo el servidor esx, carpeta, y la ubicación **resource pool** en el inventario del vCenter inventario:

```
register-VM "[london_virtualmachines] ctx-1/ctx-1.vmx" -vmhost (get-vmhost esx3.rtfm-ed.co.uk) -resourcepool (get-resourcepool London) -folder (get-folder "London VMs" | get-folder primary)
```



Nota:

Espero que lo anterior tenga sentido para usted. La razón por la que he puesto `-folder (get-folder "Reading VMs" | get-folder primary)`, es porque tengo dos carpetas, una en Londres y otra en Reading.

La salida puede ser redireccionada al `Register-VMs` cmdlet, para registrar todas las máquinas virtuales que se encuentran en ese dataStore:

```
$vmxpath = get-datastorefiles (get-datastore reading_virtualmachines) | where { $_.Path -match '.vmx$' } | select path | % { Register-VM $_.Path -vmhost (get-vmhost esx3.rtfm-ed.co.uk) -resourcepool (get-resourcepool "London VMs") -folder (get-folder "London VMs" | get-folder primary) }
```

Nota:

La única diferencia aquí es la inclusión de `$vmxpath` y él `%`. En este caso estamos haciendo la ruta de acceso a los archivos de la VMX una variable y, a continuación, enviamos el resultado al comando `Register-VM`, a fin de que cada máquina virtual (VMX ruta de archivo) descubierta en `reading_virtualmachines` VMFS es registrada en el sistema.

Recuerde que este proceso de registro, tendría que ser repetido por todos y cada uno de los volúmenes VMFS y para cada máquina virtual que necesite un proceso de recuperación.

Arreglar los archivos VMX

Usando `nano` o `vi`, en el servicio de consola, edite el archivo VMX de cada máquina virtual, para arreglar el portgroup utilizado para la comunicación.

`ethernet0.networkName = "vlan61"` tiene que ser `ethernet0.networkName = "internal"`



JmGVirtualConsulting.com

virtually anything is possible

<http://www.JmGVirtualconsulting.com>



El Blog de Virtualización en Español

Con Jose Maria Gonzalez, VCP, vExpert, RHCE, MCSE, MCSA, CPSA

<http://www.josemariagonzalez.es>

```

root@esx3:/vmfs/volumes/london_virtualmachines/ctx-1
GNU nano 1.2.1 File: ctx-1.vmx

floppy0.clientDevice = "true"
ethernet0.present = "true"
ethernet0.networkName = "vlan21"
ethernet0.addressType = "vpx"
ethernet0.generatedAddress = "00:50:56:b4:72:45"
chipset.onlineStandby = "false"
guestOSAltName = "Microsoft Windows Server 2003, Enterprise Edition"
guestOS = "winnetenterprise"
uuid.bios = "50 2a 13 b4 9a d5 8a 87-90 be c3 d3 15 bb d"

```

Por supuesto, tiene que repetir esto por cada máquina virtual.

Si usted añade primero su máquina virtual en el vCenter (nuestra próxima tarea), puede automatizar el cambio de propiedad (como he mencionado anteriormente) con PowerShell para VMware.

get-vm | get-networkadapter | sort-object -property "NetworkName" | where {'vlan21' -contains \$_.NetworkName} | Set-NetworkAdapter -NetworkName "testBubble-1 group"

```

PS C:\> get-vm | get-networkadapter | sort-object -property "NetworkName" | where {'vlan21' -contains $_.NetworkName} | Set-NetworkAdapter -NetworkName "testBubble-1 group"

Confirm
Are you sure you want to perform this action?
Performing operation "Setting NetworkName: testBubble-1 group" on Target "Network Adapter 1".
[Y] Yes [A] Yes to All [N] No [L] No to All [S] Suspend [?] Help (default is "Y"): A

MacAddress      : 00:50:56:aa:46:21
WakeOnLanEnabled : True
NetworkName     : testBubble-1 group
ConnectionState : VMware.VimAutomation.Client20.ConnectInfoImpl
ID              : VirtualMachine-vm-1742/4000
Name            : Network Adapter 1

MacAddress      : 00:50:56:aa:18:43
WakeOnLanEnabled : True
NetworkName     : testBubble-1 group
ConnectionState : VMware.VimAutomation.Client20.ConnectInfoImpl
ID              : VirtualMachine-vm-1732/4000
Name            : Network Adapter 1

MacAddress      : 00:50:56:aa:08:eb
WakeOnLanEnabled : True
NetworkName     : testBubble-1 group
ConnectionState : VMware.VimAutomation.Client20.ConnectInfoImpl
ID              : VirtualMachine-vm-1744/4000
Name            : Network Adapter 1

```

Usted podría utilizar, el comando de servicio de consola 'sed', para buscar y reemplazar la cadena del port group dentro del archivo VMX, pero creo que el método de PowerShell es más limpio. Una vez más, esta secuencia de Powershell usa get-vm y get-network para encontrar la información de cada máquina virtual y la de configuración del portgroup. Después, es ordenado y luego se filtra para mostrar sólo las máquinas virtuales con portgroup vlan61. Una vez es filtrado y es enviado al comando set-networkadapter cmdlet, el cual ajusta todas las máquinas virtuales con vlan61, sustituye el contenido del portgroup con el valor "internal".



JmGVirtualConsulting.com

virtually anything is possible

<http://www.JmGVirtualconsulting.com>



El Blog de Virtualización en Español

Con Jose Maria Gonzalez, VCP, vExpert, RHCE, MCSE, MCSA, CPSA

<http://www.josemariagonzalez.es>

Conclusiones

Como puede ver, el proceso manual es un proceso de "mano de obra" muy intensivo, lo cual es esperado por el uso de la palabra *manual*. Usted puede tener la impresión de que este problema puede ser arreglado por algún súper-script de PowerShell. Puede que incluso haya pensado, ¿por qué necesito SRM si tengo estas secuencias de comandos PowerShell?. Sin embargo, no es tan simple como parece por dos razones principales. Lo primero es que no hay realmente soporte para esta solución DR y, en segundo lugar, usted puede dedicar todo el tiempo que quiera probando sus scripts, pero luego su entorno cambiara y los scripts se quedarían obsoletos, lo que supondría un sinfín de re-ingeniería y re-análisis.

De hecho, la verdadera razón por la que quise escribir este capítulo es para mostrar cómo de "doloroso" es el proceso manual y mostrarle así los verdaderos beneficios de SRM.

Fin - Conclusiones finales

Bueno este es el final del libro, y me gustaría aprovechar esta última parte hacer algunas observaciones y conclusiones finales acerca de VMware Site Recovery Manager y VMware en general.

La primera vez que empezar a trabajar con productos VMware fue a finales del año 2003. De hecho, no fue hasta el año 2004 que me involucre seriamente con VMware ESX y VirtualCenter. Por lo tanto, veo que estamos todos ante una enorme curva de aprendizaje ya que incluso nuestros llamados expertos, gurús y evangelistas, son relativamente nuevos en la virtualización. Pero como siempre en nuestra industria, hay gente muy fuerte que trabaja en el campo que reaccionó de una forma más brillante al cambio radical que he visto venir cuando vi por primera vez una demostración de VMotion.

Se ha hablado mucho de cómo los hipervisores se están convirtiendo en un *commodity*. Todavía creo que estamos un poco lejos de llegar a esto, como demuestra la concesión de licencias de VMware - todavía hay una prima que se cobra en la capa de virtualización. Pero las cosas están cambiando y los competidores de VMware se están poniendo al día, aunque no tan rápido como a veces pensamos. Esta situación es mala para todos los interesados, incluyendo VMware. Las empresas crecen cuando tienen un mercado para crear o defender. Desde la redacción de este libro, ESX3i se ha convertido en un producto libre, por lo que ahora podría argumentar que el hipervisor es un *commodity*. Sin embargo, esto significa que ahora el dinero se ha trasladado a la gestión y SRM está firmemente en ese campamento.

Pero veo un cambio que igualmente es sísmico, en cuanto a que ha producido una verdadera revolución en nuestras herramientas de gestión, porque, sencillamente, las antiguas herramientas de gestión, simplemente no estaban a la altura. No son conscientes de las máquinas virtuales. Es una pena que vmware.com ya está registrado, sino fuera así, podría haber sido el nuevo nombre de RTFM!

VMware está creando productos que tienen en cuenta las máquinas virtuales (Lab Manager, State Manager, Site Recovery Manager, LifeCycle Manager y Virtual Desktop Manager). Así que si es usuario de VMware, no espere más y súbese a jugar con estas tecnologías, como lo he hecho yo, porque son "la próxima gran tecnología" que siempre ha estado buscando en su carrera.



JmGVirtualConsulting.com

virtually anything is possible

<http://www.JmGVirtualconsulting.com>



El Blog de Virtualización en Español

Con Jose Maria Gonzalez, VCP, vExpert, RHCE, MCSE, MCSA, CPSA

<http://www.josemariagonzalez.es>

[root@josemariagonzalez.es]# FIN

