Управление виртуальными машинами Hyper-V с помощью PowerShell

winitpro.ru/index.php/2021/11/10/upravlenie-vm-hyper-v-powershell

itpro

Статья посвящена особенностям управления виртуальными машинами Hyper-V из консоли PowerShell. Мы рассмотрим создание виртуальных коммутаторов и виртуальных машин, изменение настроек ВМ и управление ими. Вы сможете использовать рассмотренные команды для ручного управления своими ВМ или в PowerShell скриптах для автоматизации различных задачей.

Установка роли Hyper-V в Windows Server и Windows 10

Для установки роли Hyper-V хост должен иметь процессор, поддерживающий виртуализацию со SLAT. В Windows Server для установки роли Hyper-V используется команда:

```
Install-WindowsFeature -Name Hyper-V -IncludeManagementTools -Restart
```

В десктопных редакциях (Windows 10 и 11) роль Hyper-V устанавливается так:

```
Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Hyper-V –All
```

Для управления хостом Hyper-V на компьютере должен быть установлен модуль Hyper-V. Полный список команд в модуле (зависит от версии Windows) можно вывести так:

```
Get-Command -Module hyper-v
```

```
Cmdlet          Suspend-VM                                         2.0.0.0    hyper-v
Cmdlet          Suspend-VMReplication                              2.0.0.0    hyper-v
Cmdlet          Test-VHD                                           2.0.0.0    hyper-v
Cmdlet          Test-VMNetworkAdapter                              2.0.0.0    hyper-v
Cmdlet          Test-VMReplicationConnection                       2.0.0.0    hyper-v
Cmdlet          Update-VMVersion                                   2.0.0.0    hyper-v
Cmdlet          Wait-VM                                            2.0.0.0    hyper-v


PS C:\Users\Administrator> Get-Command -Module hyper-v |measure


Count    : 245
Average  :
```

В Windows Server 2022 в модуле Hyper-V доступно 245 командлетов.

Вывести полный список настроек хоста Hyper-V можно с помощью команды:

```
Get-VMHost|fl *
```

Чтобы вывести только информацию о количестве доступных ядер и RAM:

```
Get-VMHost| select LogicalProcessorCount, MemoryCapacity
```

```
PS C:\Windows\system32> Get-VMHost| select LogicalProcessorCount, MemoryCapacity

LogicalProcessorCount MemoryCapacity
--------------------- --------------
                   48    68619681792
```

Чтобы изменить настройки хоста Hyper-V используется командлет Set-VMHost. Следующая команда изменит пути по-умолчанию для хранения виртуальных дисков и конфигурационных файлов ВМ:

```
Set-VMHost -VirtualMachinePath D:\VM -VirtualHardDiskPath 'D:\VM\VHD'
```

Создаем виртуальный коммутатор Hyper-V с помощью PowerShell

Прежде всего на сервере Hyper-V нужно создать виртуальный коммутатор. Виртуальные машины смогут получать доступ к сети только через виртуальный коммутатор.

Выведем список доступных физических адаптеров на хосте Hyper-V:

```
Get-NetAdapter | where {$_.status -eq "up"}
```

Если ваш сервер поддерживает **SR-IOV (Single-Root Input/Output (I/O) Virtualization)**, обратите внимание, что нужно включать эту опцию во время создания коммутатора. Включить SR-IOV для существующего vSwitch нельзя. Более подробно это описано в статье Включаем поддержку SR-IOV для виртуальных машин Hyper-V.
Создайте виртуальный внешний коммутатор:

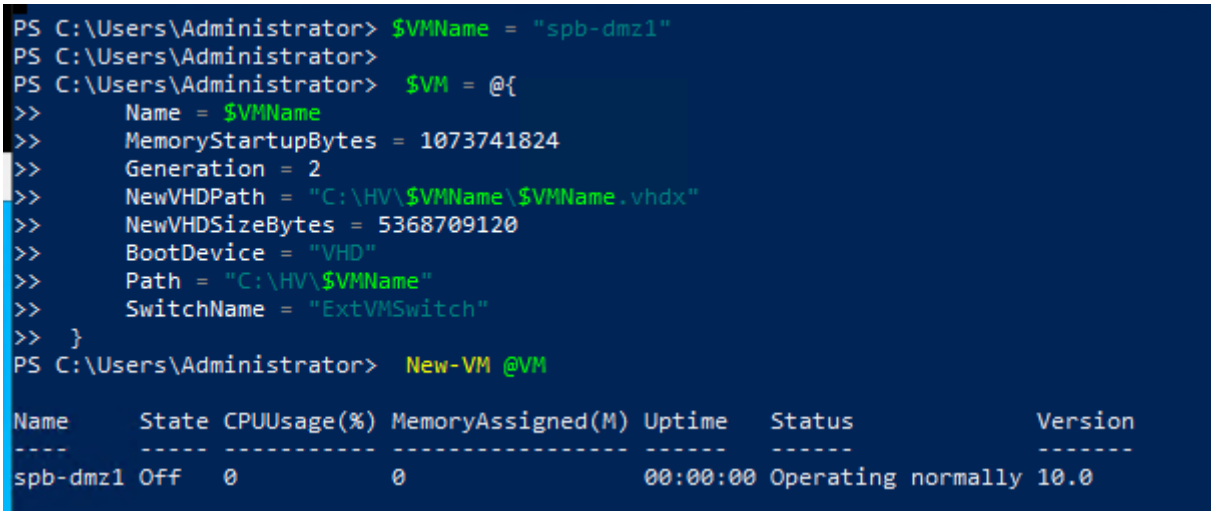
```
New-VMSwitch -Name "ExtVMSwitch" -AllowManagementOS $True -NetAdapterName
Ethernet0 -SwitchType External
```

```
PS C:\Users\Administrator> Get-NetAdapter | where {$_.status -eq "up"}

Name                      InterfaceDescription                    ifIndex Status       MacAddress             LinkSpeed
----                      --------------------                    ------- ------       ----------             ---------
Ethernet0                 Intel(R) 82574L Gigabit Network Conn...       9 Up           00-0C-29-86-41-E3         1 Gbps
PS C:\Users\Administrator> (get-vmhost).IovSupport
False
PS C:\Users\Administrator> New-VMSwitch -Name "ExtVMSwitch" -AllowManagementOS $True -NetAdapterName Ethernet0

Name        SwitchType NetAdapterInterfaceDescription
----        ---------- ------------------------------
ExtVMSwitch External   Intel(R) 82574L Gigabit Network Connection
```

Создание и изменение настроек виртуальной машины Hyper-V с помощью PowerShell

Для создания новой виртуальной машины используется командлет New-VM. В этом примере мы создадим новую ВМ второго поколения с 1 ГБ RAM и vhdx диском размером 5 Гб.

```
$VMName = "spb-dmz2"
$VM = @{
Name = $VMName
MemoryStartupBytes = 1Gb
Generation = 2
NewVHDPath = "C:\HV\$VMName\$VMName.vhdx"

NewVHDSizeBytes = 5Gb
BootDevice = "VHD"
Path = "C:\HV\$VMName"
SwitchName = "ExtVMSwitch"
}
New-VM @VM
```



Пример использования PowerShell для создания виртуальной машины с Windows 11 на Hyper-V.
Рассмотрим команды, которые можно использовать для изменения настроек виртуальных машин.

Увеличить размер RAM для ВМ:

```
Get-VM -Name spb-dmz1| Set-VMMemory -StartupBytes 2Gb
```

Изменить количество vCPU:

```
Set-VMProcessor spb-dmz1 -Count 2
```

Разрешить автозапуск для виртуальной машину Hyper-V:

```
Get-VM –VMname spb-dmz1 | Set-VM –AutomaticStartAction Start
```

Чтобы подключить дополнительный виртуальный диск в ВМ, нужно сначала создать его:

```
New-VHD -Path 'C:\VM\test1.vhdx' -SizeBytes 2GB
```

А затем подключить к ВМ:

```
Add-VMHardDiskDrive -VMName spb-dmz1 -Path 'C:\VM\test1.vhdx'
```

Используем PowerShell для управления виртуальными машинами Hyper-V

Вывести список виртуальных машин на хосте Hyper-V:

```
Get-VM
```

```
PS C:\Windows\system32> Get-VM

Name      State CPUUsage(%) MemoryAssigned(M) Uptime   Status             Version
----      ----- ----------- ----------------- ------   ------             -------
spb-app01 Off   0           0                 00:00:00 Operating normally 9.0
spb-db01  Off   0           0                 00:00:00 Operating normally 9.0
Spb-dc03  Off   0           0                 00:00:00 Operating normally 9.0
```

Команда вернула список ВМ с несколькими базовыми характеристиками. Чтобы вывести все свойства ВМ, выполните:

```
Get-VM -Name spb-dmz1 | fl *
```

Вывести только включенные ВМ:

```
Get-VM | where {$_.State -eq 'Running'}
```

Запустить виртуальную машину:

```
Start-VM -Name spb-app01
```

Запустить все выключенные виртуальные машины:

```
Get-VM | where {$_.State -eq 'Off'} | Start-VM
```

Выключить ВМ (корректное выключение через гостевую ОС):

```
Stop-VM -Name spb-app01
```

Чтобы выключить ВМ по питанию используется ключ TurnOff:

```
Stop-VM -Name spb-app01 –TurnOff
```

Зависшие ВМ можно выключить так.
Подключить ISO файл в виртуальное CD/DVD устройство:

```
Set-VMDvdDrive -VMName spb-app01 -Path c:\iso\WinSrv2022.iso
```

Прямой проброс USB диска с хоста Hyper-V в виртуальную машину описан в статье https://winitpro.ru/index.php/2014/06/26/kak-napryamuyu-probrosit-usb-disk-v-virtualnuyu-mashinu-hyper-v/

Чтобы перенести все файлы ВМ на лету на другой диск, используйте команду:

```
Move-VMStorage spb-app01 -DestinationStoragePath D:\VM\spb-app01
```

Увеличить или сжать виртуальный диск можно с помощью команды Resize-VHD:

```
Resize-VHD -Path 'C:\VM\fs01.vhdx' -SizeBytes 50Gb
```

Создать чекпоинт (снапшот) указанной ВМ:

```
Get-VM -Name spb-app01| Checkpoint-VM -SnapshotName "before install patch"
```

Вывести список доступных чекпоинтов:

```
PS C:\Users\Administrator> Get-VM -Name spb-app01| Checkpoint-VM -SnapshotName "before install patch"
PS C:\Users\Administrator> Get-VM -Name spb-app01|Get-VMCheckpoint

VMName    Name                SnapshotType CreationTime          ParentSnapshotName
------    ----                ------------ ------------          ------------------
spb-app01 before install patch Standard     11/10/2021 4:44:52 AM
```

Вернуть состояние ВМ из предыдущему чекпоинту:

```
Restore-VMCheckpoint -Name "before install patch" -VMName spb-app01 -
Confirm:$false
```

Удалить снапшот:

```
Remove-VMCheckpoint -VMName spb-app01 -Name "before install patch"
```

Экспорт, импорт и клонирование ВМ описаны подробно в статье по ссылке:

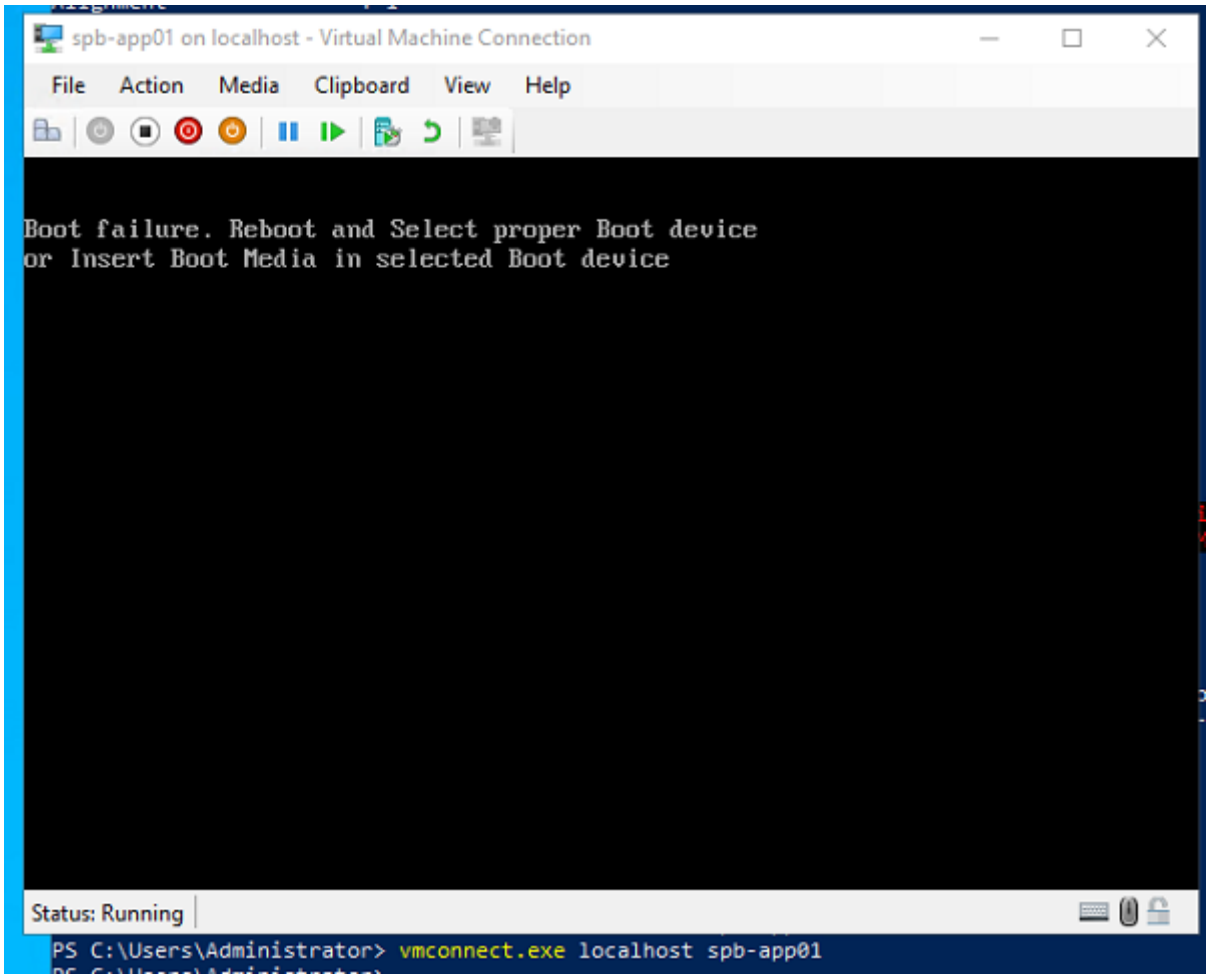
```
Export-VM -Name spb-app01 -Path 'C:\VHD\export' -CaptureLiveState
CaptureCrashConsistentState
```

Для резервного копирования виртуальных машин Hyper-V можно использовать встроенный Windows Server Backup.
Получить IP адреса гостевых ОС виртуальных машин:

```
Get-VM | Select -ExpandProperty NetworkAdapters | Select VMName,
IPAddresses, Status
```

Подключиться к консоли определенной виртуальной машины:

```
vmconnect.exe localhost spb-app01
```



Для подключения PowerShell сессией напрямую к гостевым ОС виртуальных машин через шину vmbus можно использовать PowerShell Direct (доступен для гостевых ОС Windows Server 2016, Windows 10 и новее). Можно использовать командлеты Invoke-Command (для запуска скриптов) и Enter-PSSession (для входа в интерактивную PowerShell сессию):

```
Invoke-Command -VMName spb-app01 -ScriptBlock {Get-Process}
Enter-PSSession -VMName spb-app01
```

Для копирования файлов с хоста Hyper-V в виртуальную машину через PowerShell Direct используйте:

```
$PSSession1 = New-PSSession --VMName spb-app01 -Credential (Get-Credential)
Copy-Item -ToSession $PSSession1 -Path C:\iso\win10.iso -
Destination D:\ISO\
```

Вы можете использовать PowerShell для локального или удаленного управления виртуальными машинами на хостах Hyper-V (как на Windows Server в режимах Full GUI или Core, так и на Free Windows Hyper-V Server, или Windows 10) как отдельно, так и в дополнении к графическим средствам управления Hyper-V Manager и Windows Admin Center.

1. 

   Александр 11.11.2021
   Для создания новой виртуальной машины используется командлет **Nev-VM** исправьте

   ```
   itpro: исправлено
   ```

   Ответить

2. 

   Павел 16.03.2022
   Здравствуйте.
   А есть ли возможность добавления физического жесткого диска к ВМ через powershell?

   Ответить

   itpro 22.03.2022
   Посмотрите справку командлета `Add-VMHardDiskDrive`
   _https://docs.microsoft.com/en-us/powershell/module/hyper-v/add-vmharddiskdrive?view=windowsserver2022-ps
   Там как раз есть такие примеры.

   Ответить

3. 

Eugene 21.04.2022
Можно ли из гостя узнать имя хоста Hyper?

Ответить

itpro 21.04.2022
если на ВМ стоят Integration Services, поглядитие в реестре
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Virtual Machine\Guest\Parameters
HostName
PhysicalHostName
PhysicalHostNameFullyQualified

Ответить