

Monitor Hyper-V with the command line (Part 3): Check virtual machine status

The previous article showed you how to turn on and off the virtual machine with the command line. This article will introduce two methods of validating that virtual machines are running and responding.

TipsMake.com - The previous article showed you how to turn on and off the virtual machine with the command line. This article will introduce two methods of validating that virtual machines are running and responding.

Note that the user commands do not belong to Hyper-V. They are in the Hyper-V monitoring library. moreover, this library must be imported into PowerShell each time you plan to use any Hyper-V related commands. The library entry statement is:

```
Import-Module 'C:\Program Files\Modules\Hyperv\Hyperv.psd1'
```

The full instructions on how to download and install the Hyper-V monitoring library are provided in Part 1 of this series.

Error finding commands

Administrators often depend on the ping command to test the server's connection. Very simple. Perform a ping command to the machine and see if it responds. Although the ping command works relatively well, there is a command in PowerShell called Ping-VM that is also specifically designed to ping virtual machines.

First of all, the Ping-VM command seems redundant and the syntax is a bit more complicated than the usual ping command, but the command has its own advantages. The ping-VM command requires the user to provide the virtual machine name to be pinged and the name of the host server. For example, if we want to ping a virtual machine called Lab-DC that is running on a name server called Hyper-V, the command structure will be:

```
Ping-VM 'Lab-DC' -Server Hyper-V
```

When this command is executed, the Ping-VM command uses the integrated services on the virtual machine to find Fully Qualified Domain Name and then it performs an ICMP ping command.

There are two reasons to use Ping-VM instead of Ping. First, Ping-VM recognizes the virtual machine. Second, Ping-VM can run multiple virtual machines using wildcards. To illustrate these concepts, let's look at the picture below. The host server has several virtual machines (Lab-). We will execute the order for these virtual machines. Ping-VM returned information on 6 virtual machines and could even determine which virtual machines were turned off.

```
Administrator: Windows PowerShell
PS C:\Users\Administrator.PRODUCTION> ping-vm lab-* -server "Hyper-U"

UMElementName      : Lab-E2K10
FullyQualifiedDomainName : Lab-E2K10.lab.com
NetworkAddress      : 208-87-35-103.securehost.com
ResponseTime        : 198
ResponseTimeToLive  : 49
StatusCode           : 0
Status              : Success

UMName              : Lab-W7B
FullyQualifiedDomainName :
NetworkAddress      :
Status              : UM Stopped

UMName              : Lab-W7
FullyQualifiedDomainName :
NetworkAddress      :
Status              : UM Stopped

UMElementName      : Lab-DC
FullyQualifiedDomainName : LAB-DC.lab.com
NetworkAddress      : 208-87-35-103.securehost.com
ResponseTime        : 83
ResponseTimeToLive  : 49
StatusCode           : 0
Status              : Success

UMName              : Lab-W7C
FullyQualifiedDomainName :
NetworkAddress      :
Status              : UM Stopped

UMName              : Lab-SharePoint
FullyQualifiedDomainName :
NetworkAddress      :
Status              : UM Stopped

PS C:\Users\Administrator.PRODUCTION> _
```

You can also use Ping-VM to check the status of virtual machines located on host machines. To do this, simply replace the virtual machine name with an asterisk. You can even do more and display a warning message for any virtual machine running, but not ping. The statement is:

```
Get-VM -r | foreach-object {if ((Ping-VM $ _). StatusCode -ne 0) {'$ ($ _ . elementname) is inaccessible'}}
```

The command does not output if the virtual machine is successfully pinged.

In addition, the Ping-VM command is also used to ping virtual machines on multiple Hyper-V hosts. Just give the name of each host you want to ping. Machine names must be separated by commas. For example, if you want to ping all virtual machines on Hyper-V1, Hyper-V2 and Hyper-V 3, the command to use is:

```
Ping-VM * -Server Hyper-V1, Hyper-V2, Hyper-V3
```

Check virtual machine status

Not only can Ping-VM be used to see if a virtual machine responds, users can also use the Test-VmHeartBeat command. Like Ping-VM, the Test-VMHeartBeat command depends on the integrated services running on the virtual machine tested.

The command syntax is relatively simple. The user must specify the virtual hostname and duration. Duration is calculated in seconds. The Test-VmHeartBeat command will check the status every 5 seconds until the specified time runs out. For example, If you want to check the status of the virtual machine for 5 minutes, set the time limit to 300. Observe the picture below to see more clearly for Lab-DC virtual machines.

```

Administrator: Windows PowerShell
PS C:\Users\Administrator.PRODUCTION> Test-VMHeartBeat "Lab-dc" -Timeout 300
ElementName                                     Status
-----
Lab-DC                                          OK
PS C:\Users\Administrator.PRODUCTION>

```

Like the Ping-VM command, users can use the Wildcard of the virtual machine name and assign multiple server servers.

Since Ping-VM and Test-VMHeartBeat are both used to test virtual machine responses, users can wonder which command to use.

If you want a quick status report from all virtual machines (even those that don't work), it's better to use Ping-VM. The reason is that Test-VMHeartBeat will block and wait for the time limit for each virtual machine that is currently disabled. For example, suppose the server server has 10 virtual machines, but only 5 of them are active now. We execute the following command:

```
Test-VMHeartBeat * -Timeout 300
```

The command will take up to 25 minutes to complete because the Test-VM command will wait for a full 5 minutes while checking each virtual machine is turned off. In contrast, the Ping-VM command will provide similar basic status information, but will complete almost immediately.

```

Administrator: Windows PowerShell
PS C:\Users\Administrator.PRODUCTION> Test-VMHeartBeat "Lab-dc" -Timeout 300

Waiting for heartbeat
Lab-U7
No Heartbeat component found

ElementName                                     Status
-----
Lab-E2K10                                       OK
Windows Server 8                               OK
Lab-U7B                                        No Heartbeat component found
-

```

However, Test-VMHeartBeat is especially useful when virtual machines need to be started in turn. For example, Lab-DC needs to be launched before Exchange Server (Lab-E2K10). Use the Test-VMHeartBeat command to check if Lab-DC is running before Exchange Server. To start these two servers sequentially, we can use the following command:

```
Start-vm 'Lab-DC'; Test-VMHeartBeat 'Lab-DC' -Timeout 300; Start-vm 'Lab-E2K10'
```

```

Administrator: Windows PowerShell
PS C:\Users\Administrator.PRODUCTION> Start-VM "Lab-DC" ; Test-VMHeartBeat "Lab-DC" -Timeout 300; Start-VM "Lab-E2K10"
WARNING: The job to Change state of VM Lab-DC to Running is still running in the background.

Waiting for heartbeat
Lab-DC
No Heartbeat component found

```

In the image above, users can see PowerShell start Lab-DC and then stop to wait for the virtual machine status to be recorded. After Lab-DC works, Exchange Server is started as shown below.

```

Administrator: Windows PowerShell
PS C:\Users\Administrator.PRODUCTION> Start-VM "Lab-DC" ; Test-VMHeartBeat "Lab-DC" -Timeout 300; Start-VM "Lab-E2K10"
WARNING: The job to Change state of VM Lab-DC to Running is still running in the background.
You can check its progress with Test-vmJob or Test-vmJob -statusOnly using the following job id:
\\SERVER-0\root\Virtualization\Noun_ConcreteJob.InstanceID="1DF06688-F037-4C88-88B2-384D9F3F0E31"
JobStarted
ElementName                                     Status
-----
Lab-DC                                          OK
WARNING: The job to Change state of VM Lab-E2K10 to Running is still running in the background.
You can check its progress with Test-vmJob or Test-vmJob -statusOnly using the following job id:
\\SERVER-0\root\Virtualization\Noun_ConcreteJob.InstanceID="84D4A4C8-7759-4F18-8B06-6C5D1E249B1"
JobStarted
PS C:\Users\Administrator.PRODUCTION>

```

Conclude

Now, you know how to check virtual machine status with the command line. In the next section, we will see some tricks for allocating memory for virtual machines.

You finished reading the article "**Monitor Hyper-V with the command line (Part 3): Check virtual machine status**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.