Windows 10

# Powershell

注释符号：#

**变量**

变量以 $ 符号开头

$a = "This is a string" #变量赋值

$b = 123

$c = 0.125

Set-Variable var 100

get-variable var #获取单个变量值

$var

clear-variable var #清空变量值

remove-variable var #删除变量

#强制指定变量类型：[int] 、[long]、[string] 、[char] 、[bool] 、[byte] 、[double] 、[decimal] 、[single]、[array] 数组对象、[xml] XML对象、[hashtable] 哈希表对象即字典对象

[int] $b = 5

#常量定义

Set-Variable -name intDriveType -value 3 -option constant

**运算符**

+, -, \*, /, % #算术二元运算符：加(串联)，减，乘，除，模

=, +=, -=, \*=, /=, %= #赋值运算符

!, not, and, or #不等于，非，且，或

-eq, -ne, -gt, -ge, -lt, -le, -contains #等于, 不等于, 大于, 大于等于, 小于, 小于等于, 包含

此数组中是否包含3： 1,2,3,5,3,2 –contains 3

返回所有等于3的元素： 1,2,3,5,3,2 –eq 3

返回所有小于3的元素： 1,2,3,5,3,2 –lt 3

测试 2 是否存在于集合中： if (1, 3, 5 –contains 2)

# 变量的方法

$date = Get-Date #获取当前时间

$date.AddDays(3) #当前时间加三天

& 可用于调用脚本块或者命令/函数的名称

$a.substring(0,3) # . 可用于方法调用

[DateTime]::IsLeapYear(2008) # :: 用于静态方法调用

.. 范围运算符

$a -is [int] #-is 类型鉴别器

1 -as [string] #-as 类型转换器

**字符串**

$a = "This is the 1st string"

$b = "This is the 2nd string"

$c = $a + " and " + $b # 连接两个字符串变量

+ 连接两个字符串

\* 按指定次数重复字符串

-f 设置字符串格式

-replace 替换运算符 用法："abcd" -replace "bc","TEST" 返回结果：aTESTd

-match 正则表达式匹配

-like 通配符匹配

**数组**

$var=1..6 #定义数组

foreach($i in $var) { #循环遍历数组对象

$n++

Write-Host "$i"

}

$aryComputers = "loopback", "localhost"

$arrName[0]

foreach ($strComputer in $aryComputers)

$n = 0

"a","b","c","d" | foreach {

$n++

Write-Host $\_

}

$n = 0

while($n -le 5) {

write-host $n

$n++

}

if（条件1）{ 处理1 }

elseif（条件2）{处理2}

else {处理3}

如：

Get-Service |foreach{

if($\_.status -eq "running"){

write-host $\_.displayname "("$\_status")" -foregroundcolor "green"

}

else {

write-host $\_.displayname "("$\_status")" -foregroundcolor "red"

}

}

$a = 3

switch($a) {

1 {"It's one";break}

2 {"It's two";break}

3 {"It's three";break}

4 {"It's four";break}

5 {"It's five";break}

default {"It's unknown";break}

}

函数

$var1=10

function fun1{

"The Variable is $var1"

}

function fun2 {

$var1=20;

One

}

fun1

fun2

命令：

Hyper-V虚拟连接互联网

<https://blog.csdn.net/zhanghai4155/article/details/126689886>

通过内部虚拟网卡，实现虚拟机的网络访问，原理就是通过“NAT”网络（网络地址转换）

**In Windows OS**

* **Setup Hyper-V Manager**

Open powershell with admin

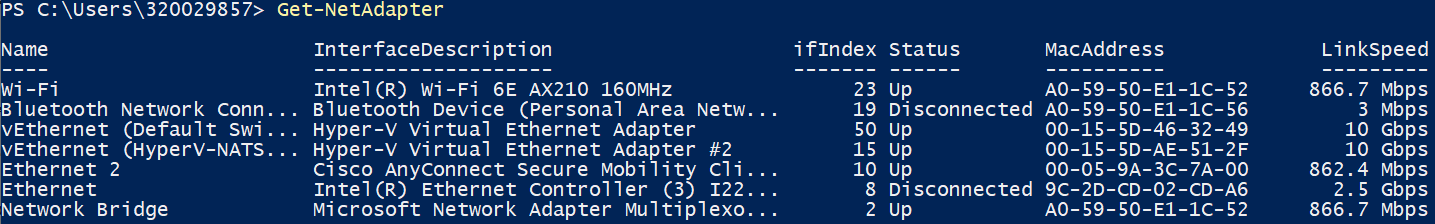
# Enable Hyper-V with a powershell admin panel

> Enable-WindowsOptionalFeature -Online -FeatureName Microsoft-Hyper-V -All

> Enable-WindowsOptionalFeature -Online -FeatureName SMB1Protocol -All

# 查看是否存在虚拟网卡（vEthernet），并记下该虚拟网卡对应的ifIndex

Get-NetAdapter



#创建虚拟交换机

New-VMSwitch -SwitchName $Natswitchname -SwitchType Internal -Verbose

Remove-NetIPAddress -InterfaceIndex 22 # 去除该虚拟网卡相关的所有NAT 网关IP设置。

Remove-NetNat # 去除所有的NAT网络配置

# 重新配置虚拟网卡对应的NAT网关IP, ip地址为192.168.0.1，子网掩码长度为24

New-NetIPAddress -IPAddress 192.168.0.1 -PrefixLength 24 -InterfaceIndex 22

# 重新构建NAT网络配置

New-NetNat -Name MyNATnetwork -InternalIPInterfaceAddressPrefix 192.168.0.0/24

Remove-NetNat -Name MyNATnetwork # 通过该名字去除该NAT网络配置

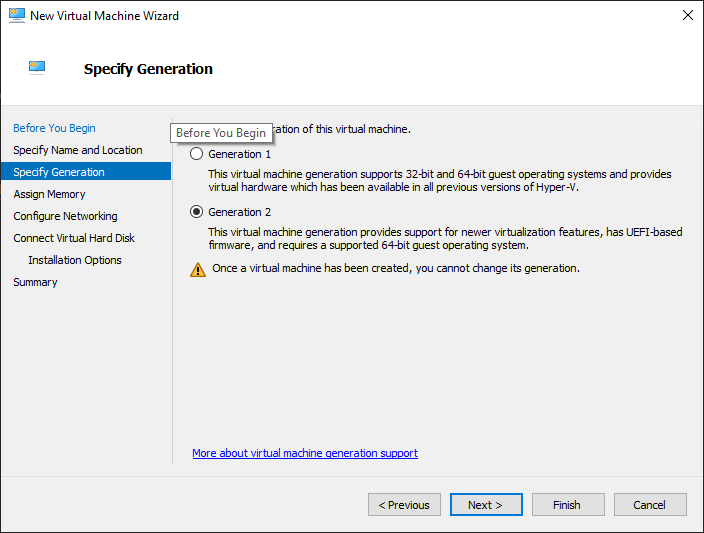
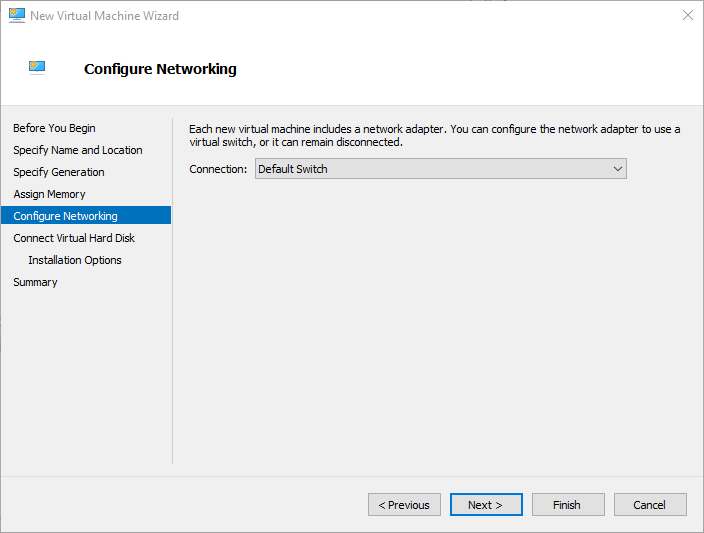
Get-NetNat -Name MyNATnetwork # 查看NAT网络配置

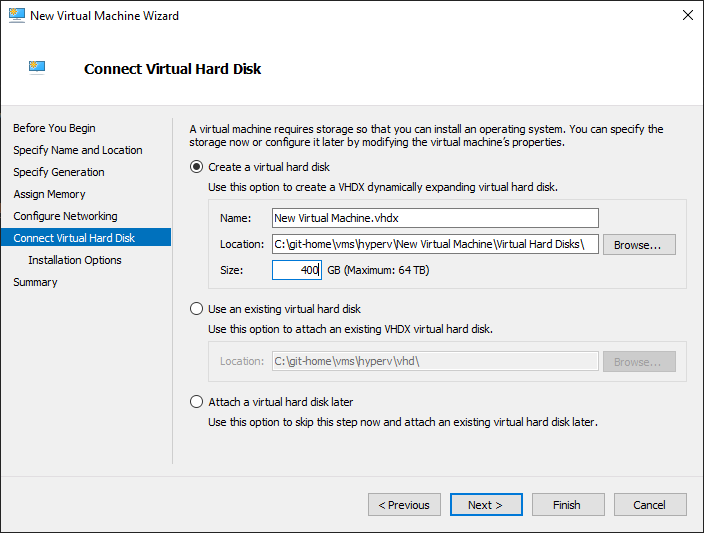
* **Bringing up the VM**

**Download the ISO** <https://artifactory-ehv.ta.philips.com/artifactory/radonc-testing-generic-local/isos/dev-vm/alma-2023-05-15.iso>

Hyper-V Manager -> Action | New | Virtual Machine

1. create a new `Generation 2` VM in the Hyper v manager
2. set the Network Switch to `Default Switch`
3. setup the Hard Drive with at least 400 GB of space



In the create VM's settings Window

1. create a 2nd network nic and assign it to the HyperV-NATSwitch
2. ensure the vm is setup to have a minimum of 4 gigs of memory available, especially if using dynamic memory as it will drop to being to low to run the installers
3. set the number of cores as appropriate for your hardware
4. disable secure boot
5. setup the DVD drive
6. Apply the current settings
7. move the 2 network boot options to the end of the list

