DAY 3 Assignment

Instructions:

Begin a new powershell session start with the below commands and provide the answers to all the questions below

```
$MyFile = (Get-Date).TimeOfDay.ToString().Replace(':','-').Replace('.','-')
$MyFile = “ReplaceThisWithYourName{1}{0}.log” -f $MyFile, ‘-’
$MyFile = Join-Path $env:USERPROFILE -ChildPath $MyFile
```

Start-Transcript -Path $MyFile

1. Using **get-wmiobject** cmdlet obtain the list of **network adapters** available on your computer
2. Obtain the help of **get-wmiobject** and refer to the **-ComputerName** property and **Write-Host** the information mentioning whether this cmdlet uses PSRemoting or Classic remoting
3. Using the appropriate cmdlets the information about the operating system using WMI and display when did the computer last booted up.Store this data in a variable **$LastBootTime**
4. Obtain the same data as in problem 3 with the corresponding CIM cmdlet and store the result in a variable **$LastBootTimeCIM**
5. Using Compare-Object check for the equality of **$LastBootTime and $LastBootTimeCIM**
6. Using the correct WMI Cmdlet supported method convert the **$LastBootTime** value to DateTime object
7. Get the type information of both **$LastBootTime** and **$LastBootTimeCIM**
8. Using **Get-Variable** cmdlet display the values of **$LastBootTime** and **$LastBootTimeCIM**
9. Using the appropriate cmdlet get the script execution permission currently set on your computer
10. Using help read about the **Get-AuthenticodeSignature** cmdlet and obtain the **SignerCertificate** property value for the file **$pshome\Certificate.format.ps1xml**
11. Write a new script file by the name **<VALIDVERB>-WMIDateTime.ps1** and implement the WmiDate conversion as in problem **6** as a script, your script should execute correctly for the below command and show results accordingly
    a. .\ **<VALIDVERB>-WMIDateTime.ps1 -wmidate $LastBootTime**
12. Using the correct cmdlet connect to your local computer as a remote host
    a. Execute **$mywsmanOpt = New-WSManSessionOption -OperationTimeout 30000**
    b. Obtain the value of the **ProxyAccessType** parameter from the results
    c. $mywsmanOpt | Get-Member and count the number of Parameters available in the **$mywsmanOpt**

Stop-Transcript

After completion please submit your Transcript Log file through email to **nanjan.suresh@gmail.com** using the **Send-MailMessage** cmdlet as an **ATTACHMENT.**