Advanced Scripting   
PowerShell and CIM

Last Updated: 6/29/2021 1:44 PM Version 1  
Document Prepared for: CIT361 Student

# Name Zach Lucas ID 895649438

# Instructions

Save a copy of this document. Answer all questions directly in this document. You will save and upload this completed document as your homework submission.

# Overview

Working with CIM is much easier in PowerShell. PowerShell provides cmdlets to find cim classes and get instances of CIM classes. It supports WMI query language as well.

# Requirements

* Windows
* PowerShell
* Remote Desktop Client software.
* Internet access.

# Task 1—Finding CIM classes

One of the keys to working with CIM is to find the correct classes that accomplish your task. PowerShell makes it pretty easy.

## Steps

1. Get all cim classes.  
   Get-CimClass
2. That’s a long list, how long is it?  
   Get-CimClass|Measure
   1. How many classes are there?1198
   2. This number should look familiar, it should be the same number you discovered with the wbemtest utility.
3. Now find a class that does something you want. For example, how much memory is in your computer. You can use wildcards to search for classes. Just like discovering cmdlets think of a good keyword then use it in your filter. This should filter you list down to a manageable scan.
4. Maybe win32\_PhysicalMemory sounds interesting.
5. Once you have found a class get it and look at it.  
   $c=Get-CimClass win32\_physicalmemory
   1. Use the CimClassProperties property to view the available properties  
      $c.CimClassProperties|Ft
      1. How many properties are there? 40
   2. Now look CimClassMethods property to view the available methods.  
      $c.CimClassMethods|Ft
      1. How many methods are there? 0
6. You try it. What class would you use to determine what version is the bios on your computer?
   1. What is the class name? Win32\_Account
   2. How many Properties does it have? 13
   3. How many Methods does it have? 0

# Task 2—Getting CIM-Instances

Once you know the class name you can use PowerShell to get an actual instance of the class with the appropriate data filled in. This is super easy.

## Steps

1. Get an instance of the Win32\_PhysicalMemory class  
   $m=Get-CimInstance win32\_physicalmemory
   1. How many instances were returned? 0
2. View the data  
   $m|ft BankLabel,Manufacturer,SerialNumber,Capacity
   1. For each instance record the BankLabel, Manufacturer, SerialNumber and Capacity  
      BankLabel = P0 CHANNEL A, P0 CHANNEL B Manufacturer=Kingston, Kingston SerialNumber=51C89724,4FC893F6 Capacity=8589934592, 8589934592

# Task 3—WMI Queries

You can use WMI queries in PowerShell as well. Let’s get a process using a query.

## Steps

1. Make sure notepad is running. Then enter the command:  
   get-ciminstance -Query "select \* from win32\_process where name='notepad.exe'"
2. You should see an instance of the notepad process.
   1. What is its WorkingSetSize? 17154048
3. Get the notepad instance again and store it in a variable;  
   $n=get-ciminstance -Query "select \* from win32\_process where name='notepad.exe'"
4. View the Cim Methods for the Win32\_Process?  
   (Get-CimClass win32\_process).CimClassMethods
   1. What methods does it have? Create, Terminate, getowner, getownersid, setpriority, attachdebugger, getavailablevirtualsize
5. Use the terminate method to stop notepad (this will stop all the instances retrieved above)  
   $n|Invoke-CimMethod -MethodName Terminate

# Task 4—CIM Remoting

CIM is designed from the very start to be remoted. To execute a CIM Query or get a CIM instance on a different machine all you must do is add a the -ComputerName parameter. For this Task you will need to connect to our lab computer.

Open your Remote Desktop software and connect to our class network using the following information.  
Computer: **cit361-lab.citwdd.net**  
User: cit361\<username> Your mailbox portion of your BYU-I email address. If your BYU-I email is [lin87690@byui.edu](mailto:lin87690@byui.edu) you would use **cit361\lin87690** for your username  
Password: Your I-Number (If you changed your password earlier in the semester you will need to use that one)

## Steps

1. Open PowerShell
2. You can use WMI to get environment information as well. Enter  
   Get-CimInstance Win32\_Environment
   1. You should see both system and user variables.
3. Use WMI remoting to get the environment from another computer  
   Get-CimInstance Win32\_Environment -ComputerName dc
4. You can pass an array of names as well  
   $computers='dc','halt'  
   Get-CimInstance Win32\_Environment -ComputerName $computers
   1. Notice the new column PSComputerName
5. If you plan getting several instances you can create CIMSessions  
   $dc=New-CimSession -ComputerName dc  
   $halt=New-CimSession -ComputerName halt  
   Get-CimInstance Win32\_Environment -CimSession $dc,$halt
6. You can see what sessions you have with  
   Get-CimSession
7. Of course, when you are done get rid of them.  
   Get-CimSession|Remove-CimSession

# Wrap-up

Logoff the lab computer

# Deliverable

Upload this document with completed answers to i-learn.