Appendix A Labs

# Review Lab 1

## Task 1

Get-EventLog –LogName Security –Newest 100

## Task 2

Get-Process | Sort –Property VM -Descending | Select –First 5

## Task 3

Get-Service | Select –Property Name,Status | Sort –Property Status –Descending | Export-CSV services.csv

## Task 4

Set-Service –Name BITS –StartupType Automatic

## Task 5

Get-ChildItem –Path C:\ -Recurse –Filter 'Win\*.\*'

## Task 6

Get-ChildItem –Path 'c:\program files' –recurse | Out-File c:\dir.txt

## Task 7

Get-EventLog –LogName Security –Newest 20 | ConvertTo-XML

## Task 8

Get-Service | Export-CSV C:\services.csv

## Task 9

Get-Service | Select –Property Name,DisplayName,Status | ConvertTo-HTML –PreContent "Installed Services" | Out-File c:\services.html

## Task 10

New-Alias –Name D –Value Get-ChildItem –PassThru | Export-Alias c:\alias.xml

*After opening a new PowerShell window…*

Import-Alias c:\alias.xml

D

## Task 11

Get-EventLog –List

## Task 12

Get-Location

## Task 13

Get-History

*After running this, locate the command that you ran for Task 11. You will need its ID number, which you will put in place of* x *in the next command:*

Get-History –id *x* | Invoke-History

## Task 14

Limit-EventLog –LogName Security –OverwriteAction OverwriteAsNeeded

## Task 15

New-Item –Name C:\Review –Type Directory

## Task 16

Get-ItemProperty -Path 'HKCU:\Software\Microsoft\Windows\CurrentVersion\Explorer\User Shell Folders'

## Task 17

* Restart-Computer
* Stop-Computer
* Remove-Computer
* Restore-Computer

## Task 18

Set-ItemProperty

# Review Lab 2

## Task 1

Get-Process | Format-Table –Property Name,ID –AutoSize

## Task 2

Get-WmiObject –class Win32\_UserAccount |   
Format-Table –Property Domain,@{n='UserName';e={$\_.Name}}

## Task 3

Invoke-Command –ScriptBlock { Get-PSProvider } –computerName *Computer1,Computer2*

## Task 4

Get-Service –computerName (Get-Content C:\Computers.txt)

## Task 5

Get-WmiObject –class Win32\_LogicalDisk –Filter "drivetype=3" |  
Where-Object { $\_.FreeSpace / $\_.Size \* 100 –gt 50}

## Task 6

Get-WmiObject –namespace root\CIMv2 –list

## Task 7

Get-WmiObject –class Win32\_Service –filter "StartMode='Auto' AND State<>'Running'

…or…

Get-WmiObject –class Win32\_Service |  
Where-Object { $\_.StartMode –eq 'Auto' –and $\_.State –ne 'Running' }

## Task 8

Send-MailMessage (read the *full* help to determine mandatory parameters)

## Task 9

Get-ACL –Path C:\

## Task 10

Get-ChildItem C:\Users | Get-ACL

## Task 11

Start-Process

## Task 12

Start-Sleep –seconds 10

## Task 13

Help \*operators\*

## Task 14

Use the Write-EventLog command

## Task 15

Get-WmiObject –class Win32\_Processor |   
Select-Object –property Manufacturer,NumberOfCores,Name,@{  
 n='MaxSpeed';e={$\_.MaxClockSpeed}}

## Task 16

Get-WmiObject –class Win32\_Process | Where { $\_.PeakWorkingSet –gt 5000 }

# Review Lab 3

1. Start-Job
2. Invoke-Command
3. Yes
4. Read-Host
5. Write-Output

## Task 1

Get-Process | Select-Object –property Name,ID,VM,PM |   
ConvertTo-HTML –Title "Current Processes" | Out-File C:\Procs.html

## Task 2

Get-Service | Export-CSV c:\services.tdf –Delimiter "`t"

## Task 3

Get-Process | Select-Object –property Name,ID,VM,PM |   
Select-Object –Property Name,ID,@{n='VM';e={$\_.VM / 1GB –as [int]}},  
 @{n='PM';e={$\_.PM / 1GB –as [int]}} |  
ConvertTo-HTML –Title "Current Processes" | Out-File C:\Procs.html