

```
1 <#
2 .SYNOPSIS
3   This module contains some common Active Directory functions that are used by many
   HSC PowerShell files.
4
5 .DESCRIPTION
6   Windows functions included in this module include:
7
8   1. Add-PSModulePath (To be written)
9   2. Get-HSCPSModulePath
10  3. Get-HSCRunAsAdministrator
11
12 .NOTES
13   HSC-WindowsModule.psm1
14   Last Modified by: Jeff Brusoe
15   Last Modified: July 22, 2020
16
17   Version: 1.0
18 #>
19
20 [CmdletBinding()]
21 [Diagnostics.CodeAnalysis.SuppressMessageAttribute("PSAvoidTrailingWhiteSpace","",Justification = "Not relevant")]
22 param ()
23
24 function Get-HSCPSModulePath
25 {
26   <#
27   .SYNOPSIS
28     This function returns the PSModulePath for the local workstation
29
30   .OUTPUTS
31     A string array of the entries for the PSModulePath variable
32
33   .EXAMPLE
34     PS C:\Users\jbrus> Get-HSCPSModulePath
35     C:\Users\jbrus\Documents\WindowsPowerShell\Modules
36     C:\Program Files\WindowsPowerShell\Modules
37     C:\WINDOWS\system32\WindowsPowerShell\v1.0\Modules
38     C:\Program Files (x86)\Microsoft SQL Server\120\Tools\PowerShell\Modules\
39     C:\HSCGitHub\HSC-PowerShell-Repository\1HSCCustomModules
40
41   .NOTES
42     Tested with the following version of PowerShell:
43     1. 5.1.18362.752
44     2. 7.0.2
45
46     Written by: Jeff Brusoe
47     Last Updated by: Jeff Brusoe
48     Last Updated: July 22, 2020
49   #>
50
51   [CmdletBinding()]
52   [OutputType([string[]])]
53   param()
54
55   try
56   {
57     [string[]]$ModulePath = (Get-ChildItem Env:\PSModulePath).Value -split ";"
58     Write-Verbose ($ModulePath | Out-String)
```

```
59
60     return $ModulePath
61 }
62 catch
63 {
64     Write-Warning "Error getting logged on user" | Out-Null
65
66     return $null
67 }
68 }
69
70 function Get-HSCRunAsAdministrator
71 {
72     <#
73     .SYNOPSIS
74         This function determines if PowerShell is running as an administrator or not.
75
76     .OUTPUTS
77         A boolean value to indicate if PS is running as an administrator
78
79     .EXAMPLE
80         PS C:\WINDOWS\system32> Get-HSCRunAsAdministrator (Running as administrator)
81         True
82
83     .EXAMPLE
84         PS C:\Users\jbrusoeadmin> Get-HSCRunAsAdministrator (Not running as
85 administrator)
86         False
87
88     .NOTES
89         Written by: Jeff Brusoe
90         Last Updated by: Jeff Brusoe
91         Last Updated: July 20, 2020
92     #>
93     [CmdletBinding()]
94     [OutputType([bool])]
95     param()
96
97     try {
98         #Taken from: https://superuser.com/questions/749243/detect-if-powershell-is-running-as-administrator
99         $RunAsAdministrator = ([Security.Principal.WindowsPrincipal]
100 [Security.Principal.WindowsIdentity]::GetCurrent()).IsInRole([Security.Principal.Wi
101 ndowsBuiltInRole]::Administrator)
102
103         Write-Verbose "Run As Administrator: $RunAsAdministrator"
104
105         return $RunAsAdministrator
106     }
107     catch {
108         Write-Warning "Unable to determine if running as administrator"
109         return $false
110     }
111 }
112
113 #####
114 # Export Functions #
115 #####
```

```
115 #Add functions
116 #Export-ModuleMember -Function "Add-HSCPSModulePath"
117
118 #Get Functions
119 Export-ModuleMember -Function "Get-*
```