



Sign in

projectHULK / AD_Recon Public

Notifications

Fork 0

Star 13

<> Code Issues Pull requests Actions Projects Security Insights

AD_Recon / SecurityAssessment.ps1



3093 lines (2742 loc) · 143 KB

Code Blame

Raw



```
1
2  <#
3      Author: Cube0x0
4      License: BSD 3-Clause
5  #>
6  Function Get-SpoolStatus {
7      <#
8      https://github.com/vletoux/SpoolerScanner
9
10         .OUTPUT
11         PS > Get-SpoolStatus -ComputerName localhost
12         ComputerName Status
13         -----
14         localhost      False
15         dc              True
16         #>
17         Param(
18             [parameter(Mandatory=$true,ValueFromPipeline=$true,ValueFromPipelineByPropertyName=$true)]
19             [string[]]$ComputerName
20         )
21         $sourceSpooler = @"
22 using System;
23 using System.Collections.Generic;
24 using System.Diagnostics;
25 using System.Runtime.InteropServices;
26 using System.Security.Permissions;
```

```
27     using System.Text;
28
29     namespace PingCastle.ExtractedCode
30     {
31         public class rprn
32         {
33             [DllImport("Rpcrt4.dll", EntryPoint = "RpcBindingFromStringBindingW",
34             CallingConvention = CallingConvention.StdCall,
35             CharSet = CharSet.Unicode, SetLastError = false)]
36             private static extern Int32 RpcBindingFromStringBinding(String bindingString, out IntPtr
37
38             [DllImport("Rpcrt4.dll", EntryPoint = "NdrClientCall2", CallingConvention = CallingConv
39             CharSet = CharSet.Unicode, SetLastError = false)]
40             private static extern IntPtr NdrClientCall2x86(IntPtr pMIDL_STUB_DESC, IntPtr formatStr
41
42             [DllImport("Rpcrt4.dll", EntryPoint = "RpcBindingFree", CallingConvention = CallingConv
43             CharSet = CharSet.Unicode, SetLastError = false)]
44             private static extern Int32 RpcBindingFree(ref IntPtr lpString);
45
46             [DllImport("Rpcrt4.dll", EntryPoint = "RpcStringBindingComposeW", CallingConvention = C
47             CharSet = CharSet.Unicode, SetLastError = false)]
48             private static extern Int32 RpcStringBindingCompose(
49             String ObjUuid, String ProtSeq, String NetworkAddr, String Endpoint, String Options
50             out IntPtr lpBindingString
51             );
52
53             [DllImport("Rpcrt4.dll", EntryPoint = "RpcBindingSetOption", CallingConvention = Callin
54             private static extern Int32 RpcBindingSetOption(IntPtr Binding, UInt32 Option, IntPtr C
55
56             [DllImport("Rpcrt4.dll", EntryPoint = "NdrClientCall2", CallingConvention = Calling
57             CharSet = CharSet.Unicode, SetLastError = false)]
58             internal static extern IntPtr NdrClientCall2x64(IntPtr pMIDL_STUB_DESC, IntPtr form
59
60             [DllImport("Rpcrt4.dll", EntryPoint = "NdrClientCall2", CallingConvention = CallingConventi
61             CharSet = CharSet.Unicode, SetLastError = false)]
62             private static extern IntPtr NdrClientCall2x64(IntPtr intPtr1, IntPtr intPtr2, stri
63
64             [DllImport("Rpcrt4.dll", EntryPoint = "NdrClientCall2", CallingConvention = Calling
65             CharSet = CharSet.Unicode, SetLastError = false)]
66             private static extern IntPtr NdrClientCall2x64(IntPtr intPtr1, IntPtr intPtr2, IntF
67
68             private static byte[] MIDL_ProcFormatStringx86 = new byte[] {
69                 0x00,0x48,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x08,0x00,0x32,0x00,0x00,0x
70                 0x00,0x04,0x00,0x08,0x00,0x00,0x48,0x00,0x00,0x00,0x00,0x01,0x00,0x
71                 0x00,0x00,0x01,0x00,0x00,0x00,0x0b,0x00,0x00,0x00,0x02,0x00,0x10,0x
72                 0x00,0x48,0x00,0x10,0x00,0x08,0x00,0x70,0x00,0x14,0x00,0x08,0x00,0x
```


110

0x00, 0x00, 0x40, 0x00, 0x00, 0x00, 0x00, 0x20, 0x00, 0x00, 0x00, 0x02, 0x00, 0x


```
3020     "@
3021         Invoke-Neo4jQuery -Query $query | Format-List -Property Neo4jData
3022     }
3023     return
3024 }
3025 return
3026 }
3027 }
3028 }
3029 }
3030 }
3031 Write-Output "Found $count weak passwords"
3032 }
3033 function ConvertFrom-CisHtml{
3034     <#
3035     .EXAMPLE
3036     PS > gci *html | foreach {ConvertFrom-CisHtml -html $_.fullname -output "C:\${$_name)}"}
3037     Found 0 improvements
3038     saved to C:\LAPTOP-CIS_Microsoft_Office_2016_Benchmark-XCCDF-.html
3039     Found 6 improvements
3040     saved to C:\LAPTOP-CIS_Microsoft_Office_Access_2016_Benchmark-XCCDF-.html
3041     Found 26 improvements
3042     saved to C:\LAPTOP-CIS_Microsoft_Office_Excel_2016_Benchmark-XCCDF-.html
3043     Found 39 improvements
3044     saved to C:\LAPTOP-CIS_Microsoft_Office_Outlook_2016_Benchmark-XCCDF-.html
3045     Found 12 improvements
3046     saved to C:\LAPTOP-CIS_Microsoft_Office_PowerPoint_2016_Benchmark-XCCDF-.html
3047     Found 19 improvements
3048     saved to C:\LAPTOP-CIS_Microsoft_Office_Word_2016_Benchmark-XCCDF-.html
3049     Found 195 improvements
```

```
3050     saved to C:\LAPTOP-CIS_Microsoft_Windows_10_Enterprise_Release_1803_Benchmark-XCCDF-.html
3051     Found 196 improvements
3052     saved to C:\LAPTOP-CIS_Microsoft_Windows_10_Enterprise_Release_1809_Benchmark-XCCDF.html
3053     #>
3054     param(
3055         [CmdletBinding()]
3056         [Parameter(Mandatory=$true, Position=0, ValueFromPipeline=$true)]
3057         [ValidateScript({Test-Path -Path $_})]
3058         $html,
3059
3060         [Parameter(Mandatory=$true, Position=1)]
3061         $output
3062     )
3063     if(get-module PSWriteWord -ListAvailable){
3064         import-module PSWriteWord
3065     }else{
3066         Write-Output "install-module PSWriteWord"
3067         throw
3068     }
3069     $html_ = Get-Content $html -Raw
3070     $rep = New-Object -com "HTMLFILE"
3071     $rep.IHTMLDocument2_write($html_)
3072     $WordDocument = New-WordDocument $output
3073     $count = 0
3074     foreach($i in ($rep.body.getElementsByClassName('Rule'))){
3075         $doc = New-Object -com "HTMLFILE"
3076         $doc.IHTMLDocument2_write(($i | select -ExpandProperty innerhtml))
3077         $res = ($doc.body.getElementsByClassName('outcome') | select -ExpandProperty outertext)
3078         if(($res) -and ($res -notmatch 'pass')){
3079             $count +=1
3080             Add-WordText -WordDocument $WordDocument -FontSize 12 -SpacingBefore 15 -Bold $true -Supress $True
3081             Add-WordText -WordDocument $WordDocument -FontSize 12 -SpacingBefore 15 -Supress $True
3082             Add-WordText -WordDocument $WordDocument -FontSize 12 -SpacingBefore 15 -Bold $true -Supress $True
3083             Add-WordText -WordDocument $WordDocument -FontSize 12 -SpacingBefore 15 -Supress $True
3084             Add-WordText -WordDocument $WordDocument -FontSize 12 -SpacingBefore 15 -Bold $true -Supress $True
3085             Add-WordText -WordDocument $WordDocument -FontSize 12 -SpacingBefore 15 -Supress $True
3086             Add-WordText -WordDocument $WordDocument -FontSize 12 -SpacingBefore 15 -Bold $true -Supress $True
3087             Add-WordText -WordDocument $WordDocument -FontSize 12 -SpacingBefore 15 -Supress $True
3088         }
3089     }
3090     $out = Save-WordDocument $WordDocument -Language 'en-US'
3091     Write-Output "Found $count improvements"
3092     Write-Output "saved to $out"
3093 }
```

