Product Solutions Resources Open Source Enterprise Pricing Sign in Sign up

This repository has been archived by the owner on Jan 21, 2021. It is now read-only.

PowerShellMafia / PowerSploit Public archive Notifications Fork 4.6k Star 11.9k

Code Issues 67 Pull requests 37 Actions Projects Security Insights

Files

d943001

Go to file

AntivirusBypass
CodeExecution
Exfiltration
Mayhem
Persistence
Privesc
Recon
Dictionaries
Get-ComputerDetail.ps1
Get-HttpStatus.ps1
Invoke-CompareAttributesForCl...
Invoke-Portscan.ps1
Invoke-ReverseDnsLookup.ps1
PowerView.ps1
README.md
Recon.psd1
Recon.psm1
ScriptModification
Tests
docs
.gitignore
LICENSE
PowerSploit.psd1
PowerSploit.psm1
PowerSploit.pssproj
PowerSploit.sln
README.md
mkdocs.yml

PowerSploit / Recon / PowerView.ps1

HarmJ0y swapped default kerberoasting output formats f94a5d2 · 6 years ago History

Code Blame Executable File · 20914 lines (15833 loc) · 752 KB Raw

```
1   #requires -version 2
2
3   <#
4
5   PowerSploit File: PowerView.ps1
6   Author: Will Schroeder (@harmj0y)
7   License: BSD 3-Clause
8   Required Dependencies: None
9
10  #>
11
12
13  ############################################################
14  #
15  # PSReflect code for Windows API access
16  # Author: @mattifestation
17  #   https://raw.githubusercontent.com/mattifestation/PSReflect/master/PSReflect.psm1
18  #
19  ############################################################
20
21  function New-InMemoryModule {
22  <#
23  .SYNOPSIS
24
25  Creates an in-memory assembly and module
26
27  Author: Matthew Graeber (@mattifestation)
28  License: BSD 3-Clause
29  Required Dependencies: None
30  Optional Dependencies: None
31
32  .DESCRIPTION
33
34  When defining custom enums, structs, and unmanaged functions, it is
35  necessary to associate to an assembly module. This helper function
36  creates an in-memory module that can be passed to the 'enum',
37  'struct', and Add-Win32Type functions.
38
39  .PARAMETER ModuleName
40
41  Specifies the desired name for the in-memory assembly and module. If
42  ModuleName is not provided, it will default to a GUID.
43
44  .EXAMPLE
45
46  $Module = New-InMemoryModule -ModuleName Win32
47  #>
48
49      [Diagnostics.CodeAnalysis.SuppressMessageAttribute('PSUseShouldProcessForStateChang
50      [CmdletBinding()]
51      Param (
52          [Parameter(Position = 0)]
53          [ValidateNotNullOrEmpty()]
54          [String]
55          $ModuleName = [Guid]::NewGuid().ToString()
```

```
        )

        $AppDomain = [Reflection.Assembly].Assembly.GetType('System.AppDomain').GetProperty
        $LoadedAssemblies = $AppDomain.GetAssemblies()

        foreach ($Assembly in $LoadedAssemblies) {
            if ($Assembly.FullName -and ($Assembly.FullName.Split(',')[0] -eq $ModuleName))
                return $Assembly
            }
        }

        $DynAssembly = New-Object Reflection.AssemblyName($ModuleName)
        $Domain = $AppDomain
        $AssemblyBuilder = $Domain.DefineDynamicAssembly($DynAssembly, 'Run')
        $ModuleBuilder = $AssemblyBuilder.DefineDynamicModule($ModuleName, $False)

        return $ModuleBuilder
    }


    # A helper function used to reduce typing while defining function
    # prototypes for Add-Win32Type.
    function func {
        Param (
            [Parameter(Position = 0, Mandatory = $True)]
            [String]
            $DllName,

            [Parameter(Position = 1, Mandatory = $True)]
            [string]
            $FunctionName,

            [Parameter(Position = 2, Mandatory = $True)]
            [Type]
            $ReturnType,

            [Parameter(Position = 3)]
            [Type[]]
            $ParameterTypes,

            [Parameter(Position = 4)]
            [Runtime.InteropServices.CallingConvention]
            $NativeCallingConvention,

            [Parameter(Position = 5)]
            [Runtime.InteropServices.CharSet]
            $Charset,

            [String]
            $EntryPoint,

            [Switch]
            $SetLastError
        )

        $Properties = @{
            DllName = $DllName
            FunctionName = $FunctionName
            ReturnType = $ReturnType
        }

        if ($ParameterTypes) { $Properties['ParameterTypes'] = $ParameterTypes }
```

```
20841         (func advapi32 LogonUser ([Bool]) @([String], [String], [String], [UInt32], [UInt32
20842         (func advapi32 ImpersonateLoggedOnUser ([Bool]) @([IntPtr]) -SetLastError),
20843         (func advapi32 RevertToSelf ([Bool]) @() -SetLastError),
20844         (func wtsapi32 WTSOpenServerEx ([IntPtr]) @([String])),
20845         (func wtsapi32 WTSEnumerateSessionsEx ([Int]) @([IntPtr], [Int32].MakeByRefType(),
20846         (func wtsapi32 WTSQuerySessionInformation ([Int]) @([IntPtr], [Int], [Int], [IntPtr
20847         (func wtsapi32 WTSFreeMemoryEx ([Int]) @([Int32], [IntPtr], [Int32])),
20848         (func wtsapi32 WTSFreeMemory ([Int]) @([IntPtr])),
20849         (func wtsapi32 WTSCloseServer ([Int]) @([IntPtr])),
20850         (func Mpr WNetAddConnection2W ([Int]) @($NETRESOURCEW, [String], [String], [UInt32]
20851         (func Mpr WNetCancelConnection2 ([Int]) @([String], [Int], [Bool])),
20852         (func kernel32 CloseHandle ([Bool]) @([IntPtr]) -SetLastError)
20853     )
20854
20855     $Types = $FunctionDefinitions | Add-Win32Type -Module $Mod -Namespace 'Win32'
20856     $Netapi32 = $Types['netapi32']
20857     $Advapi32 = $Types['advapi32']
20858     $Wtsapi32 = $Types['wtsapi32']
20859     $Mpr = $Types['Mpr']
20860     $Kernel32 = $Types['kernel32']
20861
20862     Set-Alias Get-IPAddress Resolve-IPAddress
20863     Set-Alias Convert-NameToSid ConvertTo-SID
20864     Set-Alias Convert-SidToName ConvertFrom-SID
20865     Set-Alias Request-SPNTicket Get-DomainSPNTicket
20866     Set-Alias Get-DNSZone Get-DomainDNSZone
20867     Set-Alias Get-DNSRecord Get-DomainDNSRecord
20868     Set-Alias Get-NetDomain Get-Domain
```

```
20869     Set-Alias Get-NetDomainController Get-DomainController
20870     Set-Alias Get-NetForest Get-Forest
20871     Set-Alias Get-NetForestDomain Get-ForestDomain
20872     Set-Alias Get-NetForestCatalog Get-ForestGlobalCatalog
20873     Set-Alias Get-NetUser Get-DomainUser
20874     Set-Alias Get-UserEvent Get-DomainUserEvent
20875     Set-Alias Get-NetComputer Get-DomainComputer
20876     Set-Alias Get-ADObject Get-DomainObject
20877     Set-Alias Set-ADObject Set-DomainObject
20878     Set-Alias Get-ObjectAcl Get-DomainObjectAcl
20879     Set-Alias Add-ObjectAcl Add-DomainObjectAcl
20880     Set-Alias Invoke-ACLScanner Find-InterestingDomainAcl
20881     Set-Alias Get-GUIDMap Get-DomainGUIDMap
20882     Set-Alias Get-NetOU Get-DomainOU
20883     Set-Alias Get-NetSite Get-DomainSite
20884     Set-Alias Get-NetSubnet Get-DomainSubnet
20885     Set-Alias Get-NetGroup Get-DomainGroup
20886     Set-Alias Find-ManagedSecurityGroups Get-DomainManagedSecurityGroup
20887     Set-Alias Get-NetGroupMember Get-DomainGroupMember
20888     Set-Alias Get-NetFileServer Get-DomainFileServer
20889     Set-Alias Get-DFSshare Get-DomainDFSShare
20890     Set-Alias Get-NetGPO Get-DomainGPO
20891     Set-Alias Get-NetGPOGroup Get-DomainGPOLocalGroup
20892     Set-Alias Find-GPOLocation Get-DomainGPOUserLocalGroupMapping
20893     Set-Alias Find-GPOComputerAdmin Get-DomainGPOComputerLocalGroupMapping
20894     Set-Alias Get-LoggedOnLocal Get-RegLoggedOn
20895     Set-Alias Invoke-CheckLocalAdminAccess Test-AdminAccess
20896     Set-Alias Get-SiteName Get-NetComputerSiteName
20897     Set-Alias Get-Proxy Get-WMIRegProxy
20898     Set-Alias Get-LastLoggedOn Get-WMIRegLastLoggedOn
20899     Set-Alias Get-CachedRDPConnection Get-WMIRegCachedRDPConnection
20900     Set-Alias Get-RegistryMountedDrive Get-WMIRegMountedDrive
20901     Set-Alias Get-NetProcess Get-WMIProcess
20902     Set-Alias Invoke-ThreadedFunction New-ThreadedFunction
20903     Set-Alias Invoke-UserHunter Find-DomainUserLocation
20904     Set-Alias Invoke-ProcessHunter Find-DomainProcess
20905     Set-Alias Invoke-EventHunter Find-DomainUserEvent
20906     Set-Alias Invoke-ShareFinder Find-DomainShare
20907     Set-Alias Invoke-FileFinder Find-InterestingDomainShareFile
20908     Set-Alias Invoke-EnumerateLocalAdmin Find-DomainLocalGroupMember
20909     Set-Alias Get-NetDomainTrust Get-DomainTrust
20910     Set-Alias Get-NetForestTrust Get-ForestTrust
20911     Set-Alias Find-ForeignUser Get-DomainForeignUser
20912     Set-Alias Find-ForeignGroup Get-DomainForeignGroupMember
20913     Set-Alias Invoke-MapDomainTrust Get-DomainTrustMapping
20914     Set-Alias Get-DomainPolicy Get-DomainPolicyData
```