快速通道

文档分类
- 红盟公告
- 新手上路
- 漏洞分析
- 工具介绍
- 安全配置
- 编程技术
- **安全技术** <=
- 业界动态

本类热门
- 如何利用黑客技术跟踪并分析一名目标人物
- Guest权限突破
- 有关无线安全与黑客的话题
- 让我们回归(by 默难)
- 通用ShellCode深入剖析[转]
- windows POSIX 子系统权限提升漏洞分析以及利用(v1.0)

IE IFRAM 漏洞的简单分析和临时补丁

提交文章： bkbll
提交日期： 2004-11-04
文章属性： 原创
文档类别： 安全技术
浏览次数： 今17次/总2429次

IE IFRAM 漏洞的简单分析和临时补丁
bkbll (bkbll#cnhonker.net) http://www.cnhonker.com

注: 这只是一篇临时的笔记

1. 漏洞的由来
bugtraq上有人发布了一个针对IE iframe的exploit, 用的方法非常巧妙,针对2K IE6能做到基本通用了. 牛人就是牛人.
2. 漏洞的分析.
IE 在处理iframe 标签的时候,会调用SHDOCVW!CBaseBrowser2::SetFrameName函数来进行unicode copy(wcscpy):

```
.text:71754F67 ; public: virtual long __stdcall CBaseBrowser2::SetFrameName(unsigned short *)
.text:71754F67 ?SetFrameName@CBaseBrowser2@@UAGJPAG@Z proc near ; DATA XREF: .text:717233B8 o
.text:71754F67 ; .text:71739900 o
.text:71754F67
.text:71754F67 arg_0 = dword ptr 4
.text:71754F67 arg_4 = dword ptr 8
.text:71754F67
.text:71754F67 mov eax, [esp+arg_0]
.text:71754F6B push [esp+arg_4] ; wchar_t *
.text:71754F6F add eax, 368h
.text:71754F74 push eax ; wchar_t *
.text:71754F75 call ds:__imp__wcscpy
.text:71754F7B pop ecx
.text:71754F7C pop ecx
.text:71754F7D xor eax, eax
.text:71754F7F retn 8
.text:71754F7F ?SetFrameName@CBaseBrowser2@@UAGJPAG@Z endp
```

在拷贝iframe的name时候,没有进行边界检查,导致了溢出.

3. 能覆盖的东西
这里没有进行深入研究,大部分引用exp的原文.
这个exp覆盖了一个结构,什么结构目前未知, 按照作者的说法,是在

```
7178EC02 8B08 MOV ECX, DWORD PTR [EAX]
//[0x0D0D0D0D] == 0x0D0D0D0D, so ecx = 0x0D0D0D0D.
7178EC04 68 847B7071 PUSH 71707B84
7178EC09 50 PUSH EAX
7178EC0A FF11 CALL NEAR DWORD PTR [ECX]
```

这里的时候,因为其中的一个指针被覆盖,最后当程序运行到 7178EC0A 的时候,会跳转到ecx.而ecx是一个受控制的区域.(其实
是通过暴力扩大内存区域,使0d0d0d0d总能指向我们的javascript请求的内存块) 这个从他的exp上能很好的看出来,这里就不多说了.

4. 补丁.
目前microsoft官方还没有发布补丁, 但是未了避免遭受毒害, 我还是想简单的patch一下. 思想是用msvcrt!wcsncpy来代替wcscpy.
因为要保证字节一至,就没有严格做到程序的输入/返回一样.
下面是做的简单补丁:

```
text:71754F67 sub_71754F67 proc near ; DATA XREF: .text:717233B8   o
.text:71754F67 ; .text:71739900   o
.text:71754F67
.text:71754F67 arg_0 = dword ptr 4
.text:71754F67
.text:71754F67 mov eax, [esp+arg_0]
.text:71754F6B push 20h
.text:71754F6D push esi
.text:71754F6E add eax, 368h
.text:71754F73 push eax
.text:71754F74 mov eax, 780104FCh
.text:71754F79 call eax
.text:71754F7B pop ecx
.text:71754F7C pop ecx
.text:71754F7D pop eax
.text:71754F7E nop
.text:71754F7F retn 8
.text:71754F7F sub_71754F67 endp
```

因为在前面调用的时候就是push esi来做arg 2的,所以这里直接用push esi 节省了几个字节的指令. 后面本来是要返回0的,但
因为call 一个8字节的地址在6字节内没有解决(不知道大家有没有好方法?) 所以只好牺牲eax了, 其实调用返回后eax本身也没用.

```
[root@DUMPLOGIN C:\WINNT\system32\dllcache]#fc /b shdocvw.dll shdocvw.dll.org
正在比较文件 shdocvw.dll 和 SHDOCVW.DLL.ORG
0005436B: 6A FF
0005436C: 20 74
0005436D: 56 24
0005436E: 05 08
0005436F: 68 05
00054370: 03 68
00054371: 00 03
00054373: 50 00
00054374: B8 50
00054375: FC FF
00054376: 04 15
00054377: 01 6C
00054378: 78 12
00054379: FF 70
0005437A: D0 71
0005437D: 58 33
0005437E: 90 C0
```

5. 使用.
patch补丁后, 用zap 删除掉shdocvw.dll,然后将这个copy过去,打开IE, 再来浏览exploit页,发现已经攻击无效了.

6. 郁闷:

很奇怪的是,在explorer中加载我修改后的shdocvw.dll和IE中加载的不一样,
具体在:
explorer中:

```
0:015> uf SHDOCVW!CBaseBrowser2::SetFrameName
SHDOCVW!CBaseBrowser2::SetFrameName:
00dd4f67 8b442404 mov eax,[esp+0x4]
00dd4f6b 6a20 push 0x20
00dd4f6d 56 push esi
00dd4f6e 0568030000 add eax,0x368
00dd4f73 50 push eax
```

00dd4f74 b8fc040178 mov eax,0x780104fc
00dd4f79 6760 pushad
00dd4f7b 59 pop ecx
00dd4f7c 59 pop ecx
00dd4f7d 58 pop eax
00dd4f7e 90 nop
00dd4f7f c20800 ret 0x8

IE中:
SHDOCVW!CBaseBrowser2::SetFrameName:
71754f67 8b442404 mov eax,[esp+0x4]
71754f6b 6a20 push 0x20
71754f6d 56 push esi
71754f6e 0568030000 add eax,0x368
71754f73 50 push eax
71754f74 b8fc040178 mov eax,0x780104fc
71754f79 ffd0 call eax
71754f7b 59 pop ecx
71754f7c 59 pop ecx
71754f7d 58 pop eax
71754f7e 90 nop
71754f7f c20800 ret 0x8

注意到了吗?
00dd4f79 6760 pushad
vs
71754f79 ffd0 call eax

这里没有深入研究, 不知道为什么会出现这种情况.

返回上页

Copyright © 2000-2004 **HUC** All Rights Reserved.
中国红客网络技术联盟 www.cnhonker.com
webmaster@cnhonker.net