



# 寻找客户端漏洞的艺术

图像格式中的漏洞

Venustech ADLab 赵伟



## 目标：客户端的漏洞？

- 客户端易受攻击：IE，Outlook，Firefox，MSN，Yahoo etc.
- 黑客利益的驱使：Botnet，Visa，CD-Key，DDOS etc.
- 发现漏洞较容易（More 0day?）：较容易发现，新的领域



# 为什么挖掘图像格式中的漏洞？

- Windows, Linux等操作系统支持多种图像格式：Bmp, GIF, JPG, ANI, PNG etc.文件格式众多，代码复杂易找到漏洞
- Windows中很多图像格式解析的实现方式与开源代码及其相似，经常发现同一bug☺  
(Why?)
- 黑客们并没有在每种格式中发现漏洞，没有足够的“eyes”关注



# 为什么挖掘图像格式中的漏洞？

- 从安全人员的角度：
  - 格式众多，算法复杂容易出现漏洞
  - 影响范围极广跨应用跨平台，例如：
    - Windows平台上任何解析jpg的应用，office,outlook,IE...GDIPLUS.dll
    - Windows内核实现对Ani的支持，通过ie不需要用户互动操作。谁会怀疑网页上的指针文件？
    - PNG Msn, libpng很多开源软件
  - 隐蔽性严重威胁用户安全



# 为什么挖掘图像格式中的漏洞？

## ■ 从黑客的角度：

- 如果利用图像格式触发的漏洞，会降低了受害者的警觉性，易利用社会工程学。蠕虫传播可能利用一些非常容易让人警惕的文件格式，但我们现在讨论的是图片格式jpg, png, ani...不容易让人引起怀疑
- 多种攻击媒介，利于黑客攻击：通过网页，邮件可以穿越防火墙的保护，IDS不易检查，需要对各种格式，协议进行解析才能检查出漏洞攻击。



# 图像的基本格式

- 流格式由很多段构成，段里面又由标记，参数（漏洞点），数据段构成
- 还可能有段里面再嵌套段（漏洞点）
- **Gif, Ani**可能包含很多帧，刷新率，帧的索引（漏洞点）
- 可能会有标记图形模式的**bit-map**, 可能会有逻辑上的错误**png**  
<http://scary.beasts.org/security/CESA-2004-001.txt>



# JPG格式中的漏洞

- 先来一个实际的例子：

- GDIPlus.DLL漏洞MS04-028 Nick DeBaggis

- 影响巨大，攻击很多

- 漏洞产生原因：

- JPEG格式中的注释段（COM）由0xFFFE开始(标记)+2字节得注释段字节数(参数)+注释（数据）构成。因为字节数这个参数值包含了本身所占的2字节，所以GDIPLUS.dll在解析jpg格式文件中的注释段时会把这个值减去2，如果这个值设置成0，1就会产生整数溢出。



# JPG格式中的漏洞

- 是不是觉得很相似? 😊
  - 2000 Solar Designer 发现了Netscape浏览器的JPEG解析漏洞，与Nick DeBaggis发现的漏洞原理是相同的。
  - <http://www.openwall.com/advisories/OW-002-netscape-jpeg.txt>



## 另一个相似的例子

- Stefan Esser发现的XBOX Dashboard local vulnerability, 该漏洞存在于XBOX Dashboard对.wav格式和.xtf格式文件的解析上, 虽然说不是图形格式但漏洞原理却相同。
- 细节: 同样存在一个size参数这次是它本身的大小是4字节, 所以当size值为0-3时就会发生整数溢出。



# 疑问

- 疑问：为什么会一再出现同类型的漏洞？
  - 是否程序员们从概念上忽略了某些问题？
  - 为什么都是整数溢出漏洞？
  - 此类漏洞的本质是什么？
  - 是否还有这种漏洞？



# 问题的本质

- 这些文件格式是由很多“段”构成的数据流，而每个段由：标记，参数，数据等结构构成，在程序解析这些文件格式的时候会依据“标记”来确认段，并读取“参数”进行一定的运算，再依据这些参数来处理随后紧跟的“数据”。以上提到的几个漏洞的产生原因就是在对参数进行运算的时候相信了文件输入的参数没有进行确认而导致的。



# 思维扩展

- 不要相信用户的输入，同样不要相信文件的输入☺
  - 包括标记，错误的标记也会导致问题
  - 包括参数，详细检查输入参数
  - 包括数据，数据里面可能还嵌套着另一个“段”



# 思维扩展的结果☺

- Venustech AD-Lab: Windows LoadImage API Integer Buffer overflow
- 影响极为广泛: bmp,cur,ico,ani格式的文件都受影响。
- 描述:
  - WINDOWS的USER32库的LoadImage系统API 存在着整数溢出触发的缓冲区溢出漏洞, 这个API允许加载一个bmp,cur,ico,ani格式的图标来进行显示, 并根据图片格式里说明的大小加4来进行数据的拷贝, 如果将图片格式里说明的大小设置为0xffffffffc-0xffffffff, 则将触发整数溢出导致堆缓冲区被覆盖。攻击者可以构造恶意的bmp,cur,ico,ani格式的文件, 嵌入到HTML页面, 邮件中, 发送给被攻击者, 成功利用该漏洞则可以获得系统的权限。



# LoadImage API 整数溢出漏洞分析

## ■ 代码:


```
.text:77D56178      mov     eax, [ebx+8]           //Direct read our size here:P
.text:77D5617B      mov     [ebp+dwResSize], eax
.text:77D5617E      jnz    short loc_77D56184
.text:77D56180      add     [ebp+dwResSize], 4     //add 4 int overflow...
.text:77D56184
.text:77D56184 loc_77D56184:                ; CODE XREF: sub_77D5608F+EF ↑j
.text:77D56184      push   [ebp+dwResSize]       //allocate a wrong size
.text:77D56187      push   0
.text:77D56189      push   dword_77D5F1A0
.text:77D5618F      call  ds:RtlAllocateHeap
```

总结：转换思路后找到这个加4的漏洞，同样的类型，信任“文件”输入。



# 思维扩展的结果☺


- EEYE 2004: Windows ANI File Parsing Buffer Overflow
- 堆栈漏洞极易利用，攻击方法隐蔽。
- 原理：
  - 相信“文件”输入参数，没做检查直接用作 memcopy 的参数。



# PNG漏洞，不同的模式

## ■ 逻辑问题1:

- EEYE PNG (Portable Network Graphics)  
Deflate Heap Corruption Vulnerability
- 原因：对 Length码 #286 and #287没有做正确的处理，导致解压程序认为长度是0
- `do { *dest = *src; ++dest; ++src; } while (--len);`



# PNG漏洞，不同的模式

- 逻辑问题2: libPNG 1.2.5 堆栈溢出

- 代码:

```
if (!(png_ptr->mode & PNG_HAVE_PLTE)) {  
    /* Should be an error, but we can cope with it */ png_warning(png_ptr, "Missing  
    PLTE before tRNS"); }  
    else if (length > (png_uint_32)png_ptr->num_palette) {  
png_warning(png_ptr, "Incorrect tRNS chunk length"); png_crc_finish(png_ptr,  
length); return;  
    }  
}
```

- 分析: 代码编写的逻辑错误, 错误的使用了 **else if**.

- 相似漏洞: MSN png 漏洞, Media player png 漏洞



# 逻辑问题的总结

- 非常容易出现在复杂的文件格式处理中
- 容易出现在压缩，解压代码中：需要处理很多长度，大小相关的参数。
- 这种漏洞不一定是缓冲区溢出，也可能是越界访问等等



# 想象漏洞

- 发现漏洞有时候是一种想象的过程☺
- 例子1:
  - Venustech ADLab: Microsoft Windows Kernel ANI File Parsing Crash Vulnerability
  - 介绍: ANI是WINDOWS 支持的动画光标格式, 在ANI是由多个普通的光标文件组成一个动画, 其中ANI文件的头处会标记是几个图标frame, WINDOWS 的内核在显示光标的时候并未对该值进行检查, 如果将这个数字设置为0, 会导致受影响的WINDOWS系统计算出错误的光标的地址并加以访问, 触发了内核的蓝屏崩溃。不仅仅是应用使用ANI文件时会触发, 只要在EXPLORER下打开ANI文件存在的目录就会触发。攻击者也可以发送光标的文件, 引诱用户访问含有恶意光标显示的页面, 以及发送嵌入光标的HTML邮件, 导致被攻击者系统蓝屏崩溃。
  - 原理:在计算frame地址的时候失败。



# 想象漏洞

## ■ 例子2:

- Venustech ADLab: Microsoft Windows Kernel ANI File Parsing DOS Vulnerability
- 介绍: ANI是WINDOWS 2000支持的动画光标格式,在ANI是由多个普通的光标件组成一个动画,其中ANI文件的头处会标记每FRAME切换的频率,该值越小切换的速度越快,WINDOWS的内核在切换光标FRAME的时候并未对该值进行检查,如果将这个数字设置为0,受影响的WINDOWS的内核会陷入内核的死锁,不再响应任何用户界面的操作。该漏洞触发必须要在使用ANI文件的应用中才能触发,攻击者引诱用户访问含有恶意光标显示的页面,以及发送嵌入光标的HTML邮件,导致被攻击者系统内核死锁。
- 原因: 没有考虑刷新频率是0的情况。



# 总结

- 下溢：**Size**参数小于自身所占大小
- 上溢：**Size**加上一个正整数值产生上溢
- 直接作为参数输入**memcpy**类函数
- 非法参数导致地址访问越界
- 多种逻辑上的错误
- 充分发挥想象：刷新率？ 😊



# 总结

## ■ 安全提示:

- 文件格式是攻击者的另一种输入渠道，同样不要信任从文件读取的数据
- 解析文件格式时应该对参数进行充分的检查
- 同样需要想象力，需要考虑到每种可能的情况
- 最后但同样重要：Good Luck!☺



# Thanks

安全来自未雨绸缪

Venus Info Tech Inc.

Security

Trusted {Solution} Provider

Services