密级	公开

IIS 系列 Http.sys

处理 Range 整数溢出漏洞

应急分析报告

[第一版 2015/04/16]

知道创宇安全研究团队

1. 更新情况

版本	时间	描述
第一版	2015/04/16	第一版完成。

2. 漏洞概要

2015 年 04 月 14 日，微软发布严重级别的安全公告 MS15-034，编号为 CVE-2015-1635，据称在 Http.sys 中的漏洞可能允许远程执行代码。

2.1. 漏洞描述

Http.sys 是一个位于 Windows 操作系统核心组件，能够让任何应用程序通过它提供的接口，以 Http 协议进行信息通讯。微软在 Windows 2003 Server 里引进了新的 HTTP API 和内核模式驱动 Http.sys，目的是使基于 Http 服务的程序更有效率。其实在 Windows XP 安装 SP2 后，Http.sys 已经出现在系统里了，但事实上操作系统并没有真的使用这个内核级驱动，而 XP 上自带的 IIS 5.1 也没有使用 HTTP API。

从曝出的 poc 来看，此漏洞是一个整数溢出类型的漏洞，微软安全公告称最大安全影响是远程执行代码。

2.2. 漏洞影响

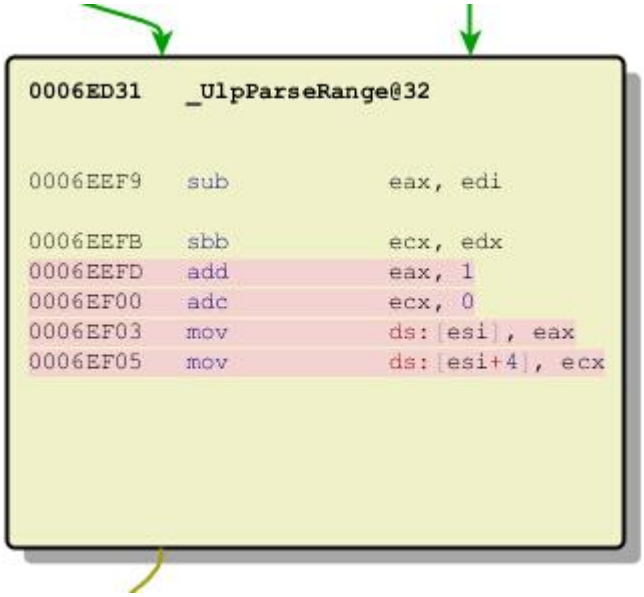
受影响版本：

IIS 7.0 以上的 Windows 7/8/8.1 和 Windows Server 2008 R2/Server 2012/Server 2012 R2 等操作系统。

2.3. 漏洞分析

根据补丁比较发现，poc 中提到的代码出现在 UlpParseRange 函数中修改的部分。

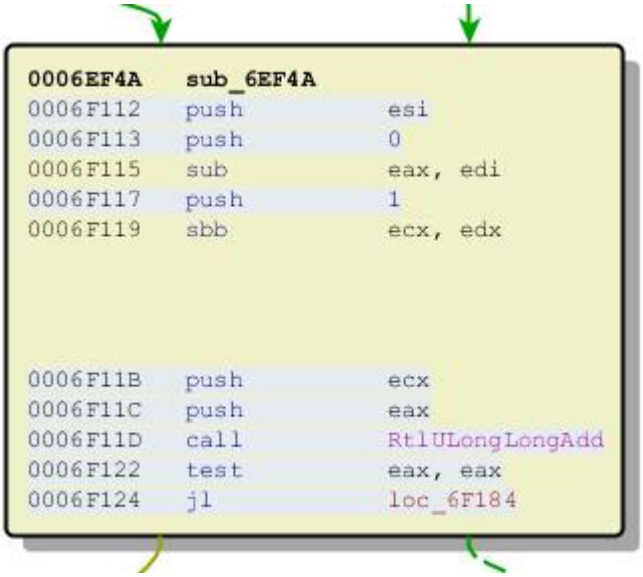
在未打补丁的 http.sys 文件的 UlpParseRange 函数中，代码如下。




```
if ( v21 < v22 || v21 <= v22 && v20 <= v23 )
  *(_QWORD *)v18 = __PAIR__(v22, v23) - __PAIR__(v21, v20) + 1;
```

可以看到，在计算 64 位整数时直接进行了运算，没有进行必要的整数溢出检查。

而在打补丁的 http.sys 文件的 UlpParseRange 函数中，修改代码如下。



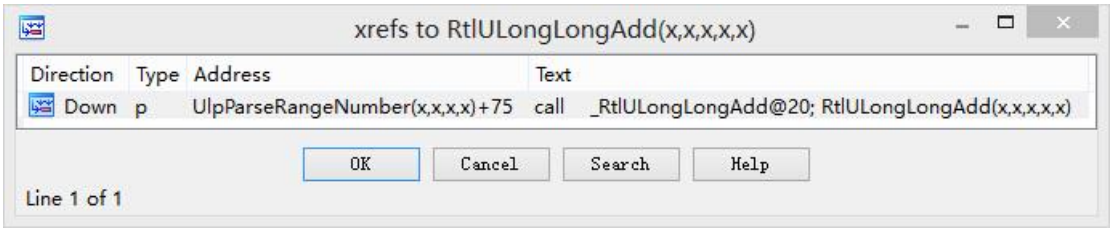

```
if ( RtlULongLongAdd(__PAIR__(v22, v23) - __PAIR__(v21, v20), 1i64, v18) >= 0 )
```

用 RtlULongLongAdd 函数来计算 Range 范围长度 v18，这个函数中是做了整数溢出检查的。

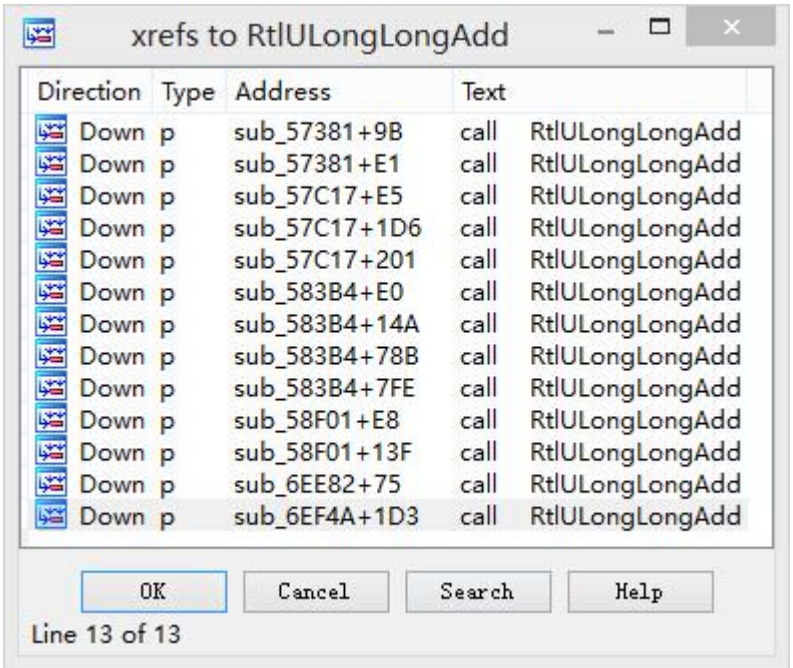
```
unsigned int __stdcall RtlULongLongAdd(__int64 a1, __int64 a2, int a3)
{
  __int64 v3; // kr00_8@1
  unsigned int result; // eax@4

  v3 = a2 + a1;
  if ( (unsigned __int64)(a2 + a1) >> 32 < HIDWORD(a1)
    || (unsigned __int64)(a2 + a1) >> 32 <= HIDWORD(a1) && (unsigned int)v3 < (unsigned int)a1 )
  {
    *(_DWORD *)a3 = -1;
    *(_DWORD *)(a3 + 4) = -1;
    result = 0xC0000095;
  }
  else
  {
    *(_QWORD *)a3 = v3;
    result = 0;
  }
  return result;
}
```

再看一下对 RtlULongLongAdd 函数的调用情况。



在未打补丁的 http.sys 文件中只有 1 处调用了 RtlULongLongAdd 函数。



而在打补丁的 http.sys 文件中总共有 13 处调用了 RtlULongLongAdd 函数进行整数溢出检查，说明有漏洞的系统中可能有多个处理流程会涉及到整数溢出造成的安全问题。

通过补丁比较确定了修改过的函数如下。

similarity	confidence	change	EA primary	name primary	EA secondary
0.23	0.44	GI--E--	0001CE31	_NLG_Notify	0001CEA1
0.25	0.44	GI--E--	0001CE50	_NLG_Call	0001CEC0
0.25	0.44	GI--E--	00051CCC	sub_51CCC_3	00051D14
0.25	0.44	GI--E--	000773B5	sub_773B5_4	000783B5
0.26	0.44	GI--E--	00014210	UlpGetProcessorNumberFromIndex(x,x)	00014282
0.82	0.97	GI-J---	000570CC	UlAdjustRangesToContentSize(x,x,x)	00057C17
0.88	0.98	GI--E--	00056EE0	UlpDuplicateChunkRange(x,x,x,x,x,x)	00057381
0.89	0.97	GI-JE-C	00058E3C	UlpBuildMultiRangeMdlChainFromSlices(x,x,x,x)	000583B4
0.90	0.98	GI-J---	000598AF	UlpBuildSingleRangeMdlChainFromSlices(x,x,x,x)	00058F01
0.98	0.99	GI--E--	00075765	GsDriverEntry(x,x)	00076765
0.99	0.99	GI-----	0006ED31	UlpParseRange(x,x,x,x,x,x,x,x)	0006EF4A

经过分析发现，UlAdjustRangesToContentSize 函数中的整数溢出点，才是导致漏洞能发挥作用的关键流程。

```
mov     edx, esi
add     edx, ecx
mov     ecx, edi
adc     ecx, ebx
cmp     ecx, dword ptr [ebp+arg_4+4]
jb      short loc_571D4
ja      short loc_571C5
cmp     edx, dword ptr [ebp+arg_4]
jb      short loc_571D4

loc_571C5:                              ; CODE XREF: UlA
                                        ; UlAdjustRanges
mov     ecx, dword ptr [ebp+arg_4]
mov     edx, dword ptr [ebp+arg_4+4]
sub     ecx, esi
sbb     edx, edi
mov     [eax+4], edx
mov     [eax], ecx
```

这段代码还是采用了直接运算 64 位整数的方式，没有检查是否溢出，在补丁文件中替换为调用 RtlULongLongAdd 函数。

这部分代码的功能是判断获取文件偏移量的范围，是否会超过请求缓存文件的数据长度，如果超出就把读取长度修改为合适的大小，防止越界访问数据。但是由于发生了整数溢出，使得判断越界的代码失效，这样就不会修改读取长度，造成用可控的长度值越界访问数据。

但是如果要成功利用此漏洞还需要一些必要的条件，具体细节有待进一步分析。

2.4. 漏洞验证

可以使用以下 python 程序对系统进行漏洞检测。

```
import socket
import random

ipAddr = "192.168.154.130"
hexAllFfff = "18446744073709551615"

req1 = "GET / HTTP/1.0\r\n\r\n"
```

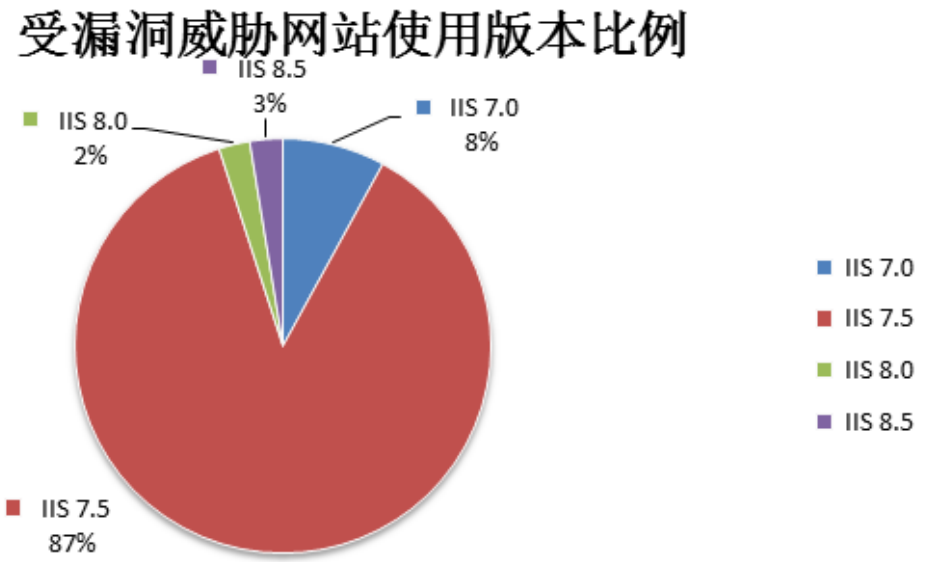
```
req = "GET / HTTP/1.1\r\nHost: stuff\r\nRange: bytes=0-" + hexAllFfff
+ "\r\n\r\n"

print "[*] Audit Started"
client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client_socket.connect((ipAddr, 80))
client_socket.send(req1)
boringResp = client_socket.recv(1024)
if "Microsoft" not in boringResp:
                print "[*] Not IIS"
                exit(0)
client_socket.close()
client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
client_socket.connect((ipAddr, 80))
client_socket.send(req)
goodResp = client_socket.recv(1024)
if "Requested Range Not Satisfiable" in goodResp:
                print "[!!] Looks VULN"
elif " The request has an invalid header name" in goodResp:
                print "[*] Looks Patched"
else:
                print "[*] Unexpected response, cannot discern patch
status"
```

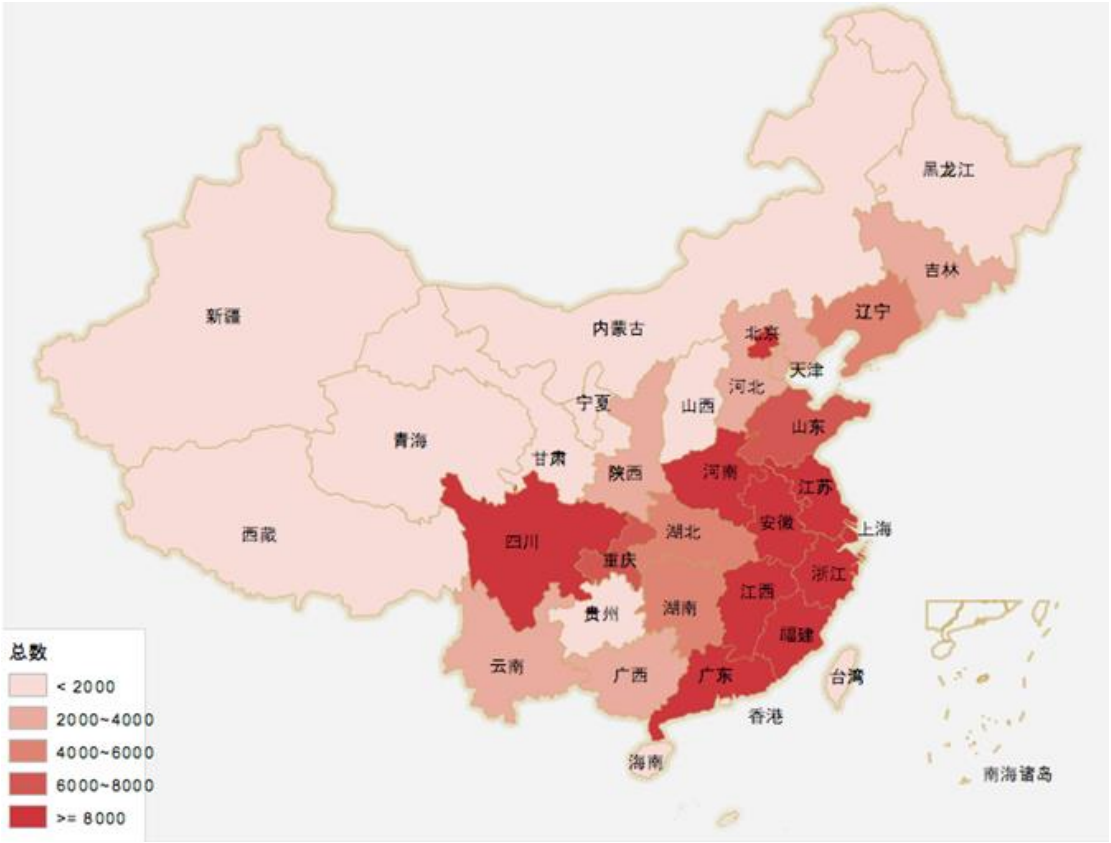
如果打印出“Looks VULN”，说明系统存在漏洞。

3. ZoomEye 应急概要

知道创宇团队通过网络空间搜索引擎 ZoomEye 进行全网搜索，得出目前网络空间中可能受影响网站所使用 IIS 版本比例如下所示：



▲受威胁网站使用版本比例



▲全国网站受 IIS 漏洞影响地域分布情况

另外，ZoomEye 搜索结果显示，全国受漏洞威胁的网站总数达 795,317 个，超过我国网站总数的五分之一，从区域分布来看，排在首位的北京地区共 276,39 个，对漏洞的修复工作刻不容缓。请网络管理员尽快打补丁修复，官方补丁下载地址：https://support.microsoft.com/zh-cn/kb/3042553。

4. 修复建议

通过 Windows 更新机制，选择 KB3042553 安全更新进行系统升级。

5. 相关资源链接

https://technet.microsoft.com/zh-cn/library/security/ms15-034