**CVE-2012-0158漏洞分析报告**

**分析人：VicZ**

2019年02月28日

目录

[1. 环境简介与poc获取 2](#_Toc7179)

[2. 漏洞复现 2](#_Toc22898)

[3. 漏洞成因 3](#_Toc25592)

[4. 利用过程 6](#_Toc32358)

# 环境简介与poc获取

操作系统：Window 7 专业版 (32 位）

软件：office 2003 sp3

工具：Ollydbg，windbg，IDA Pro，010editor，Immunity Debugger

Poc获取

网络,论坛, Metasploit

https://www.exploit-db.com/

https://www.securityfocus.com/

# 漏洞复现

打开od，设置od异常选项以及strongod插件设置，将异常设置去除勾选。

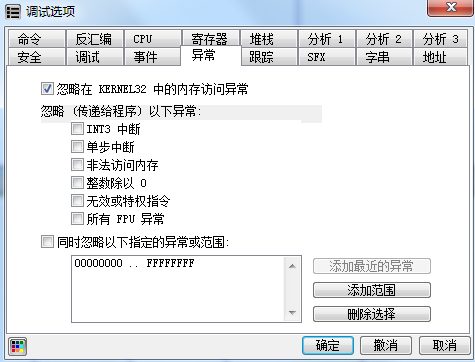


图2-1 异常设置

打开word，将word附加进od，在poc样本用word打开，发现eip指向了41414141，观察堆栈窗口，返回地址都被淹没，这可能是一个栈溢出漏洞。

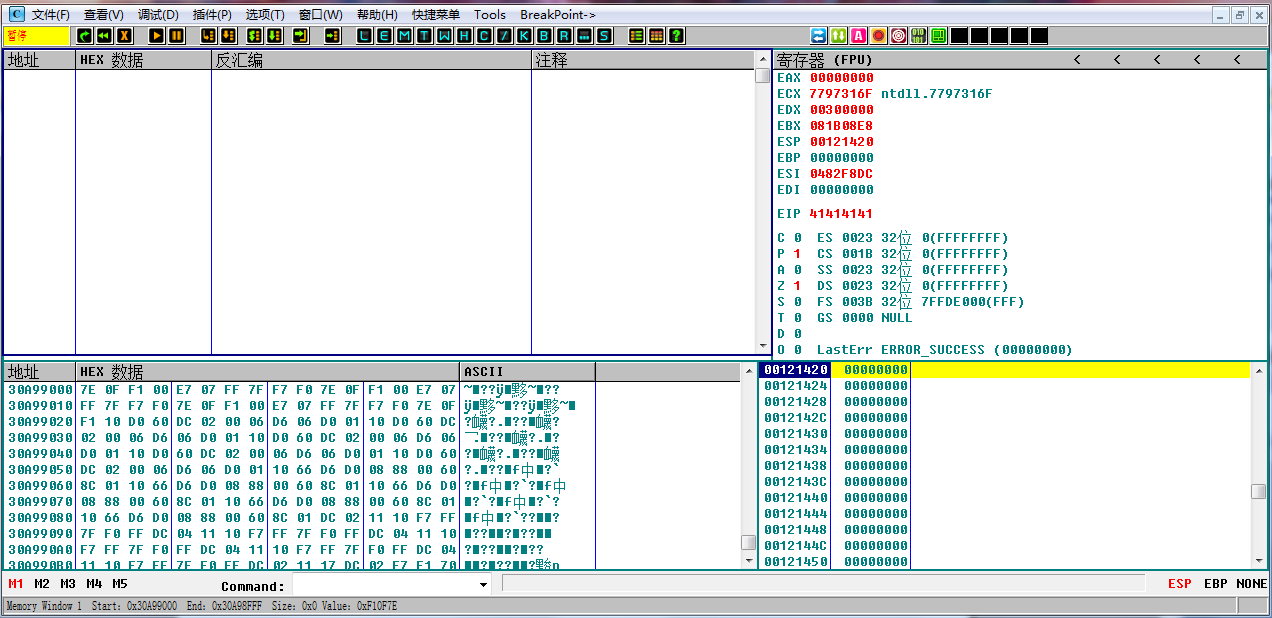


图2-2 漏洞复现

# 漏洞成因

观察堆栈，找到前一层的调用，发现是执行了275C876D这个函数才覆盖了esp

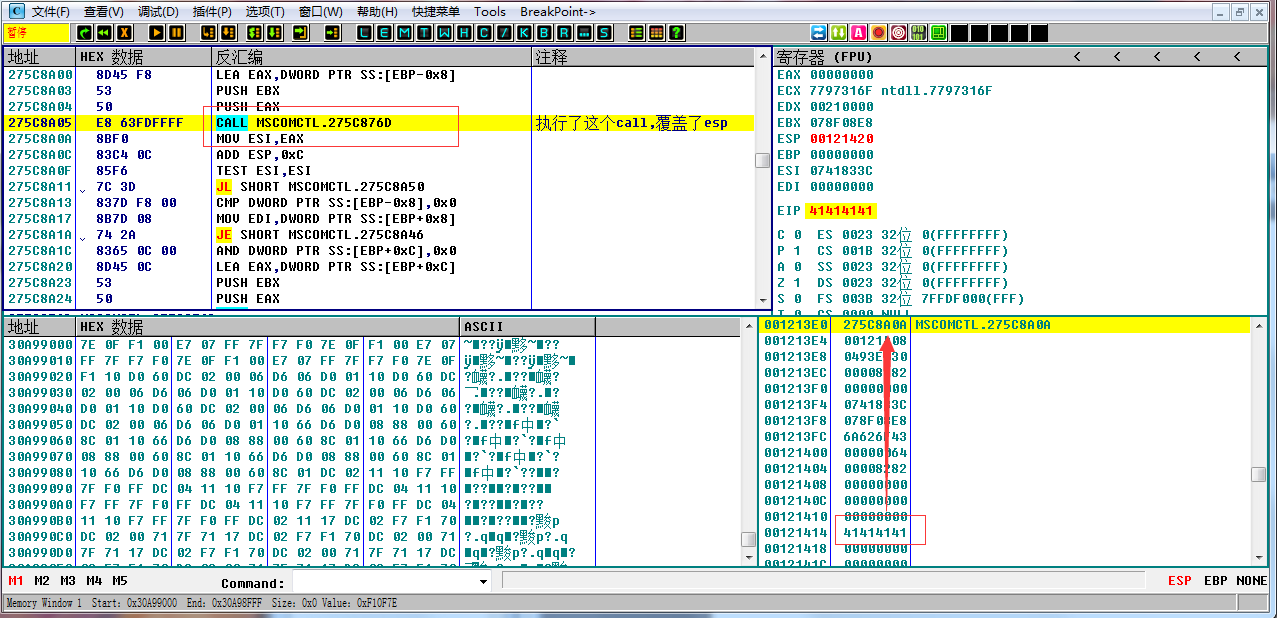


图3-1 找到关键函数

在系统中找到这个模块，用IDA进行分析。

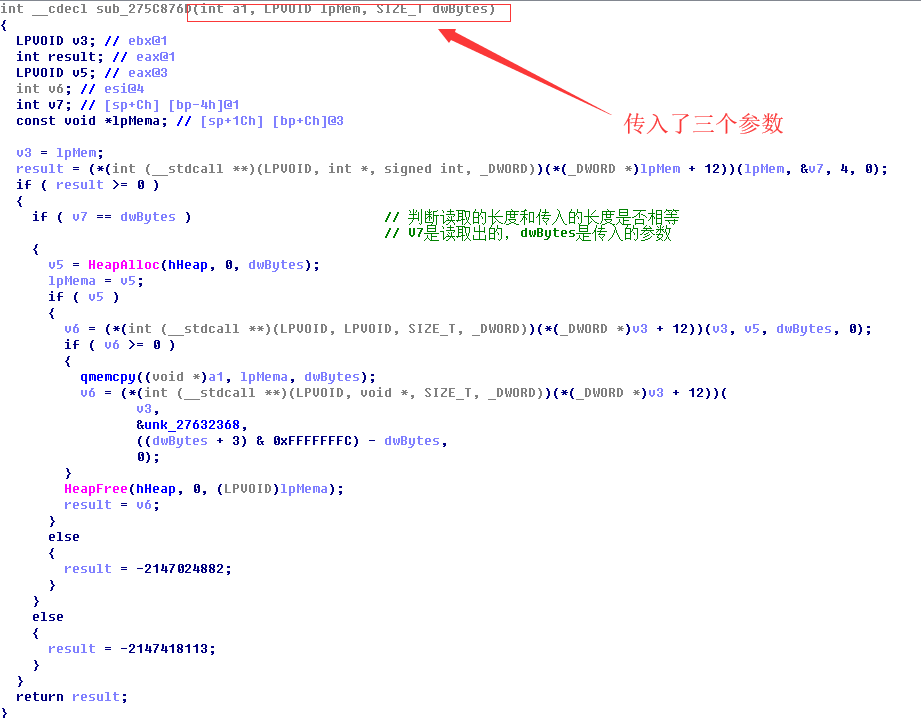


图3-2 275C876D函数

分析可知，v7和dwBytes都是从文件中读取出来的，当两个数据相等时，且他们的长度很大时，就会造成栈溢出。

观察275C876D函数传入的参数，如何3-3可知8282为传入的长度。

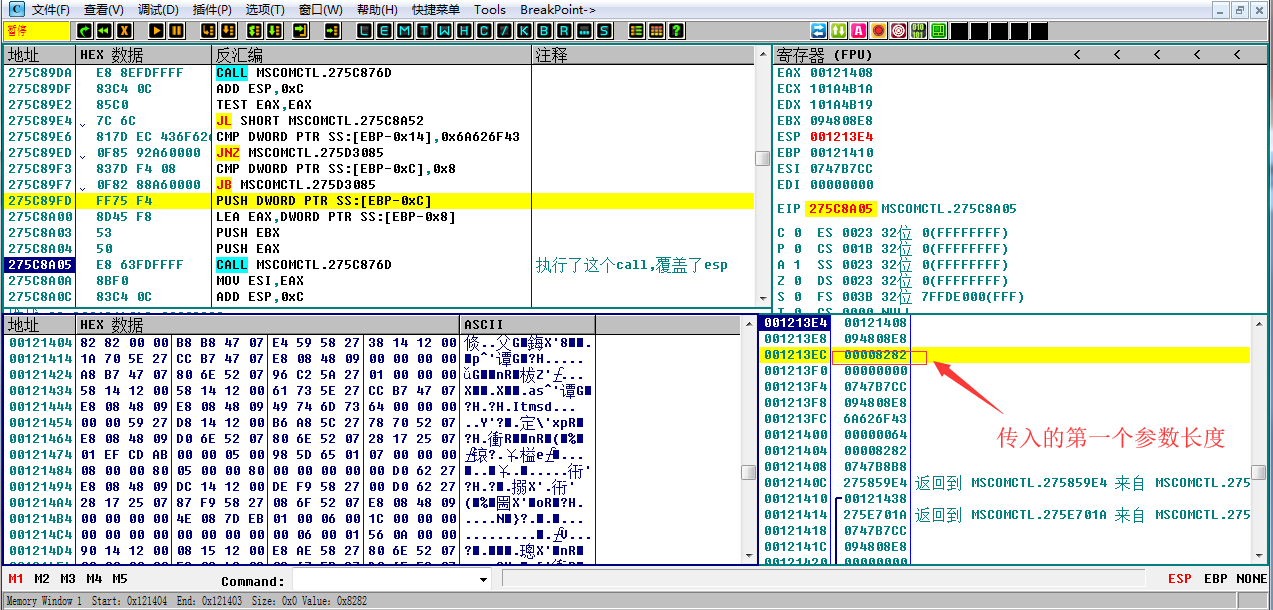


图3-3 传入参数的长度

将poc文件用010editor打开， 查找8282

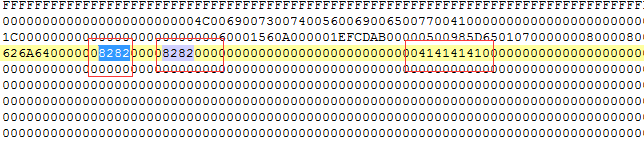


图3-4 查找8282

如图，8282有两处，猜测一个是dwBytes，一个是v7，分别修改为8283和8284

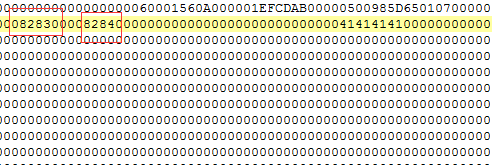


图3-5 修改数据

重复之前的步骤，再次到达传入的第一个参数，发现已经变成了8283，进入函数内部，到达dwBytes和v7的判断，v7变成了8284，说明v7和dwBytes都是从文件中读取出来的。

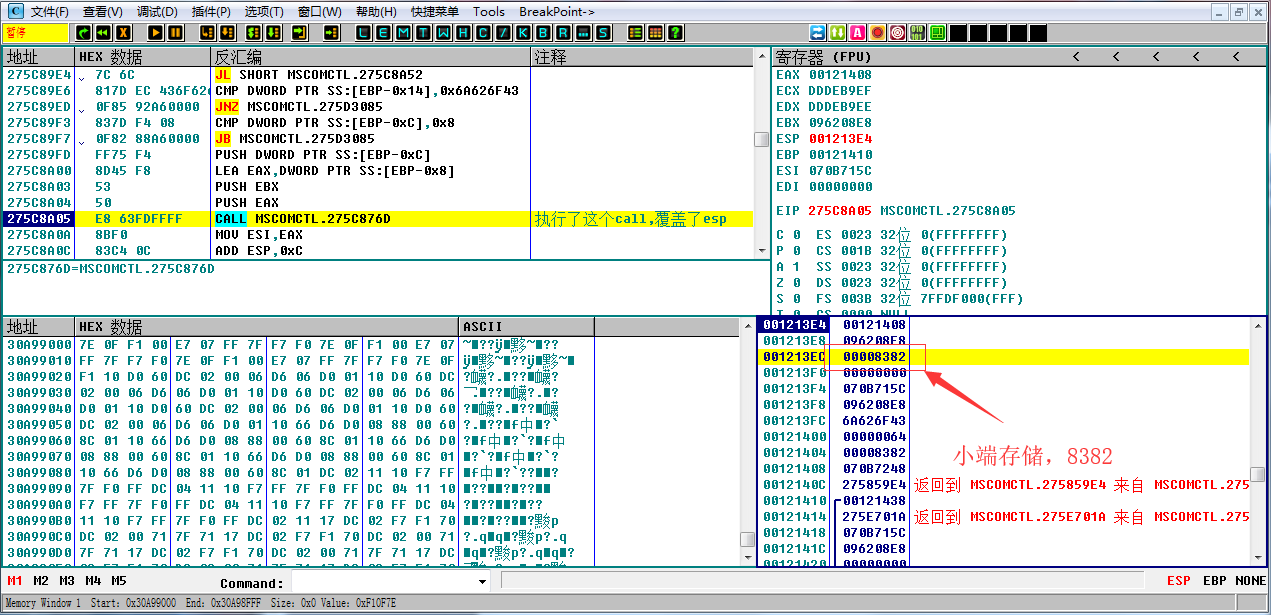


图3-6 dwBytes大小

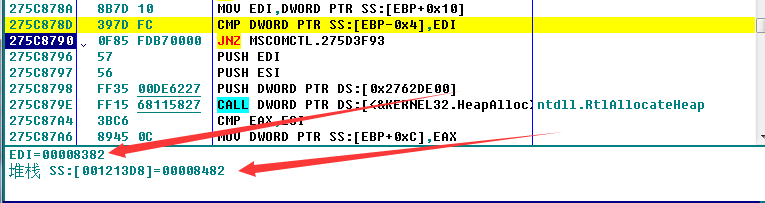


图3-7 函数内部判断

经过调试和分析之后，可以总结一下

CVE-2012-0158 漏洞触发在 MSCOMCTL.ocx 模块中，漏洞成因是在读取数据时，读取的长度和验证的长度都在文件中，所以可以自行构造，进而触发栈溢出。

# 利用过程

在poc的基础上，编写shellcode，达到利用漏洞。



图4-1 poc shellcode结构

**4.1 寻找跳板指令**

打开Immunity Debugger，附加word。利用mona脚本命令!mona moudles，查看模块

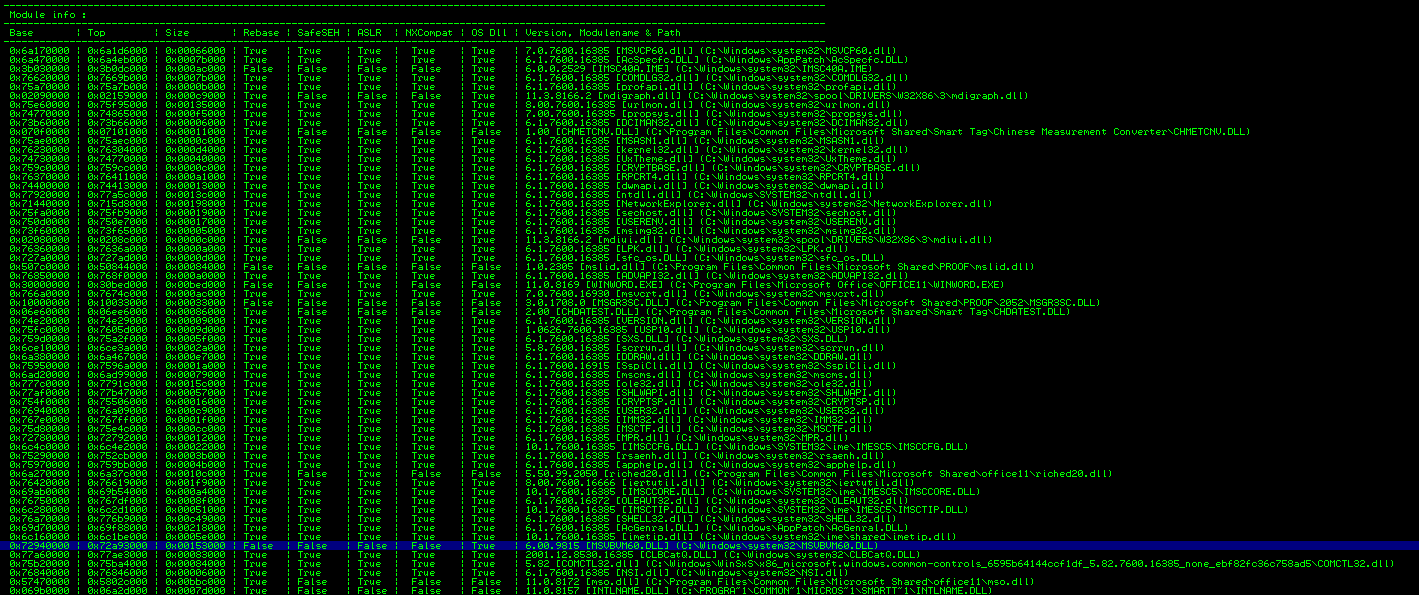


图4-2 查看模块

找到msvbvm60.dll模块，使用!mona find -s “\xff\xe4” -m msvbvm60.dll



图4-3 jmp esp

如图4-3，找到一个合适的jmp esp， 0x729A0535

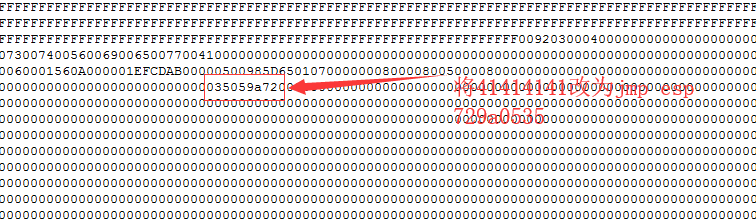


图4-4 写入跳板指令

再次重复步骤，走到尾部，eip指向了我们找的跳转地址。

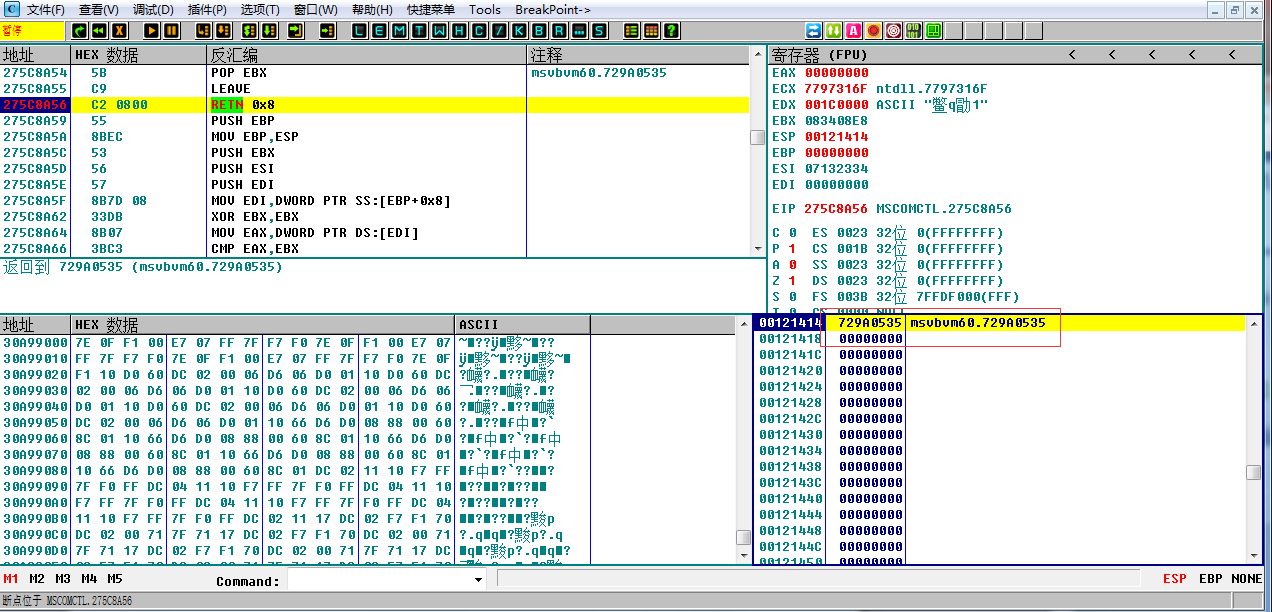


图4-5 eip指向

**4.2 编写shellcode**

根据以下的步骤，编写shellcode

1. 获取 Kernel32 基地址

2. 获取 GetProcAddress 地址

3. 获取 LoadLibraryExA 地址

4. 获取 user32 基地址

5. 获取 MessageBoxA 地址

6. 调用 MessageBoxA 地址

7. 获取 ExitProcess 地址

8. 调用 ExitProcess

首先将用到的函数名都以ASICC码形式整理出来

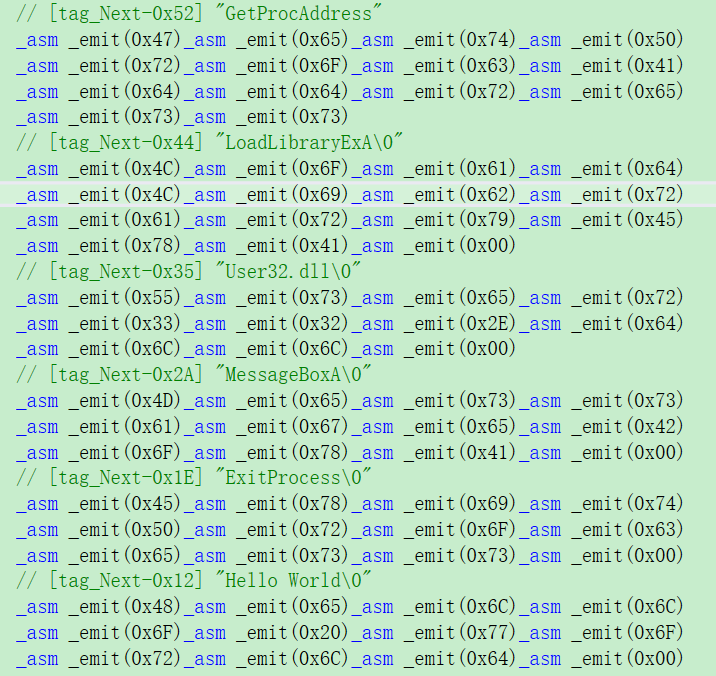


图4-6 ASICC码形式的数据

编写汇编代码，用于弹出一个hello world的MessageBox的弹框

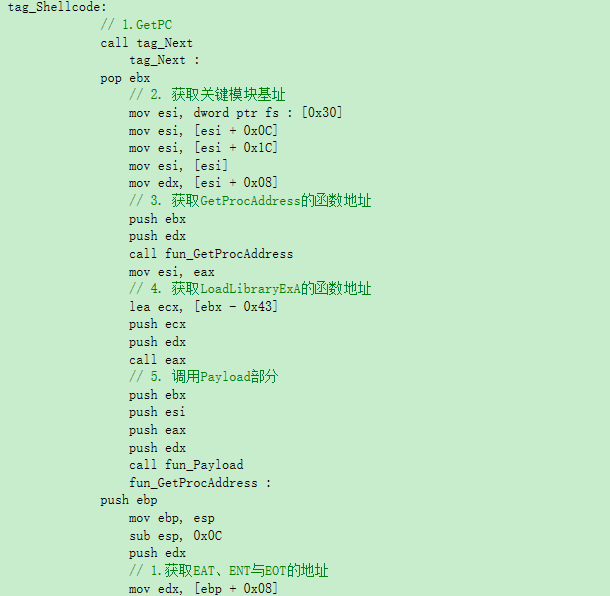


图4-7 部分汇编代码

编写成功后，测试能够弹出hello world

# 

图4-8 弹框

用od将这段shellcode提取出来复制到poc文件shellcode起点处，保存文件，打开文件，弹出hello world，漏洞利用成功。

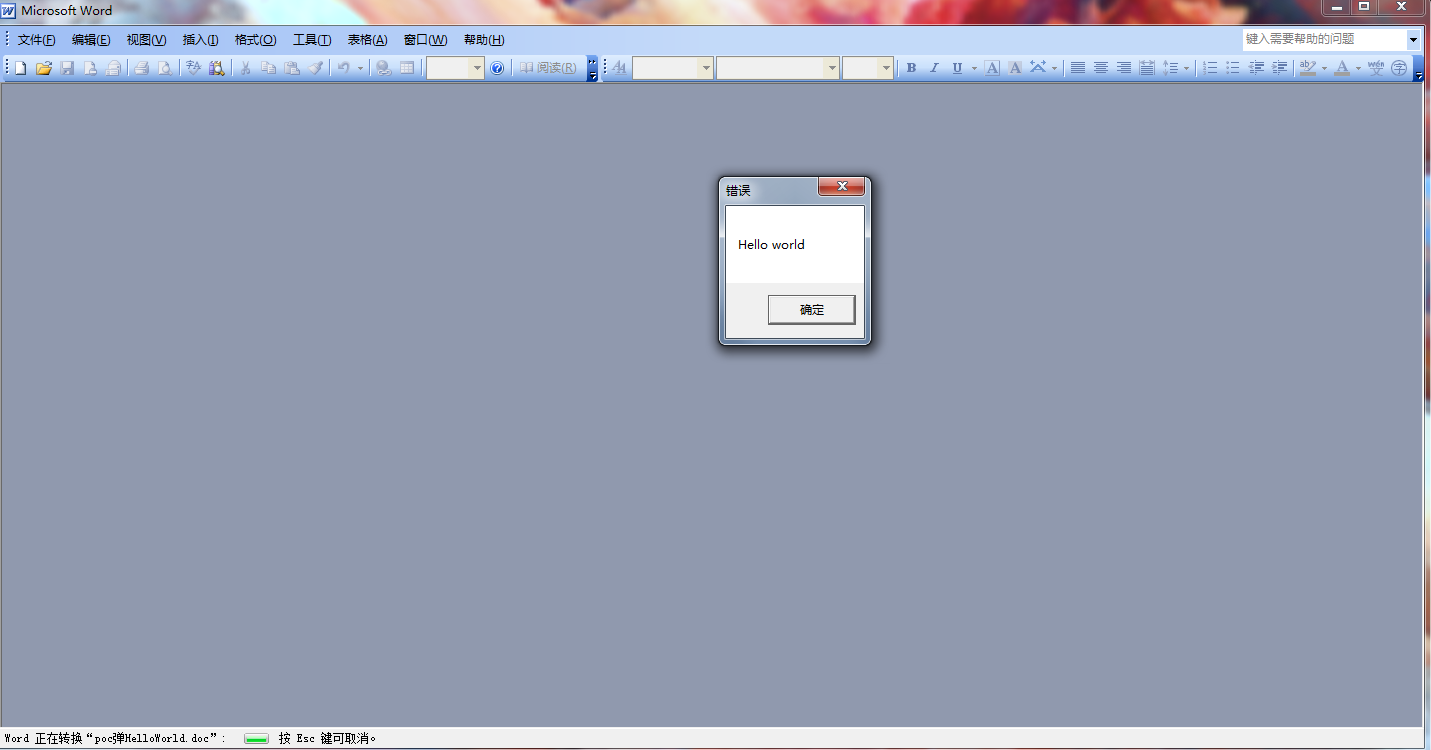


图 4-9 漏洞利用

至此，CVE-2012-0158漏洞分析完成。