

# 软件安全实验报告

姓名：郭子涵 学号：2312145 班级：信息安全、法学双学位班

## 1 实验名称：

shellcode编写及编码

## 2 实验要求：

复现第五章实验三，并将产生的编码后的shellcode在示例5-4中进行验证，阐述shellcode编码的原理、shellcode提取的思想。

## 3 实验过程：

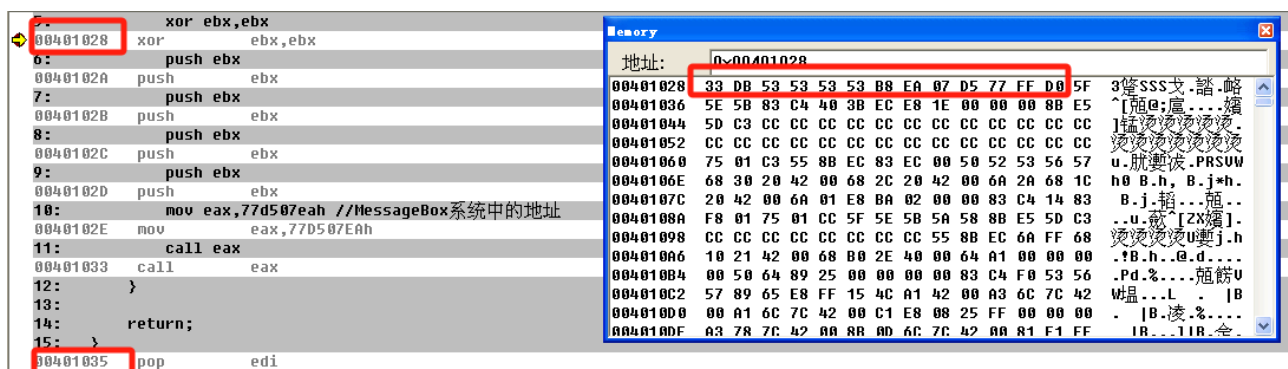
### 3.1 shellcode代码的编写和提取：

用C语言书写要执行的shellcode程序：

```
#include <stdio.h>
#include <windows.h>

void main()
{
    MessageBox(NULL,NULL,NULL,0);
    return;
}
```

换成对应的汇编代码，在代码的第一行处打断点，定位具体内存中的地址：

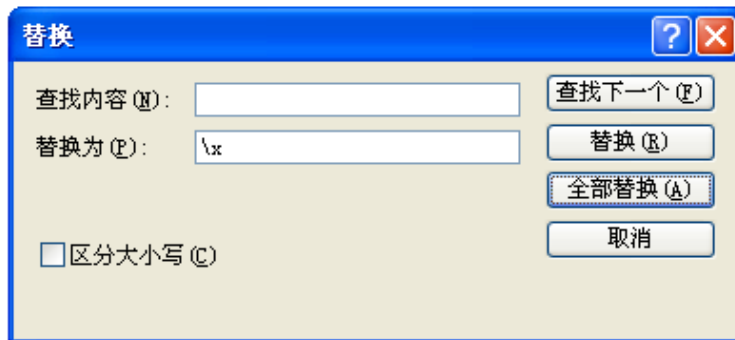


Address	Disassembly	Comment
00401028	xor ebx,ebx	
00401029	push ebx	
0040102A	push ebx	
0040102B	push ebx	
0040102C	push ebx	
0040102D	push ebx	
0040102E	mov eax,77d507eah //MessageBox系统中的地址	
0040102F	call eax	
00401030	return	
00401031	pop edi	

Address	Hex	ASCII
00401028	33 0B 53 53 53 B8 EA 07 05 77 FF D0 5F	33 0B 53 53 53 B8 EA 07 05 77 FF D0 5F
00401029	5E 5B 83 C4 40 3B EC E8 1E 00 00 00 8B E5	5E 5B 83 C4 40 3B EC E8 1E 00 00 00 8B E5
0040102A	5D C3 CC CC CC CC CC CC CC CC CC CC CC	5D C3 CC CC CC CC CC CC CC CC CC CC CC
0040102B	CC CC CC CC CC CC CC CC CC CC CC CC CC	CC CC CC CC CC CC CC CC CC CC CC CC CC
0040102C	75 01 C3 55 8B EC 83 EC 00 50 52 53 56 57	75 01 C3 55 8B EC 83 EC 00 50 52 53 56 57
0040102D	68 30 20 42 00 68 2C 20 42 00 6A 2A 68 1C	68 30 20 42 00 68 2C 20 42 00 6A 2A 68 1C
0040102E	20 42 00 6A 01 E8 BA 02 00 00 83 C4 14 83	20 42 00 6A 01 E8 BA 02 00 00 83 C4 14 83
0040102F	F8 01 75 01 CC 5F 5E 5B 5A 58 8B E5 5D C3	F8 01 75 01 CC 5F 5E 5B 5A 58 8B E5 5D C3
00401030	CC CC CC CC CC CC CC CC CC CC CC CC CC	CC CC CC CC CC CC CC CC CC CC CC CC CC
00401031	10 21 42 00 68 00 2E 40 00 64 A1 00 00 00	10 21 42 00 68 00 2E 40 00 64 A1 00 00 00
00401032	00 50 64 89 25 00 00 00 00 83 C4 F0 53 56	00 50 64 89 25 00 00 00 00 83 C4 F0 53 56
00401033	57 89 65 E8 FF 15 4C A1 42 00 83 6C 7C 42	57 89 65 E8 FF 15 4C A1 42 00 83 6C 7C 42
00401034	00 A1 6C 7C 42 00 C1 E8 08 25 FF 00 00 00	00 A1 6C 7C 42 00 C1 E8 08 25 FF 00 00 00
00401035	A3 78 7C 42 80 80 80 6C 7C 42 80 81 F1 FF	A3 78 7C 42 80 80 80 6C 7C 42 80 81 F1 FF

由上图可以看出，此段代码的地址为00401028-00401034，搜索地址可以看出对应的机器码应为：33 DB 53 53 53 53 B8 EA 07 D5 77 FF D0。利用记事本工具，用替换功能将空格转化为字节表示的方法：

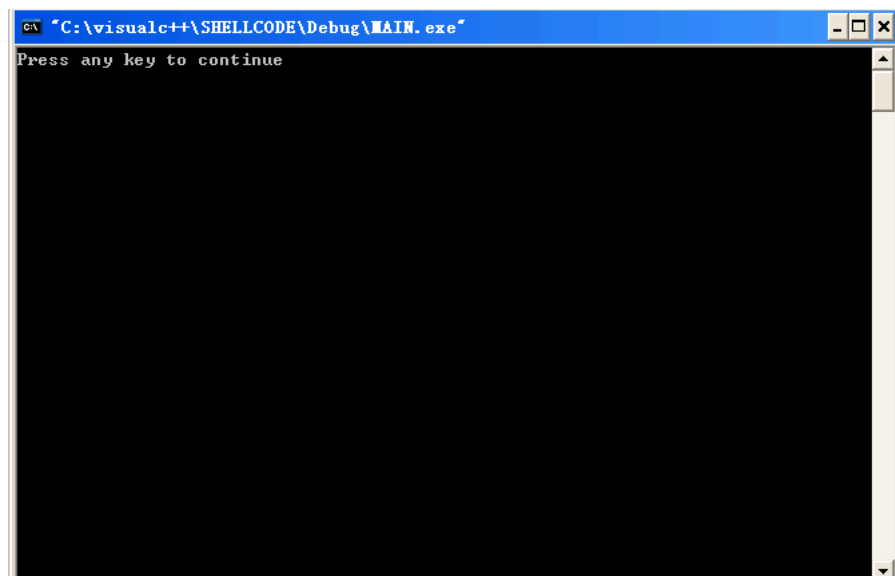
**33\xDB\x53\x53\x53\x53\xB8\xEA\x07\xD5\x77\xFF\xD0**



编写测试程序，填充shellcode的机器码：

```
#include <stdio.h>
#include <windows.h>
char ourshellcode[]="\x33\xDB\x53\x53\x53\x53\xB8\xEA\x07\xD5\x77\xFF\xD0";
void main()
{
    LoadLibrary("user32.dll");
    int *ret;
    ret=(int*)&ret+2;
    (*ret)=(int)ourshellcode;
    return;
}
```

运行程序可得下列弹窗，证明我们的shellcode代码是正确的：



## 3.2 shellcode的编码:

```
#include <stdlib.h>
#include <string.h>
#include <stdio.h>
void encoder(char* input, unsigned char key)
{
    int i = 0, len = 0;
    FILE * fp;
    len = strlen(input);
    unsigned char * output = (unsigned char *)malloc(len + 1);
    for (i = 0; i<len; i++)
        output[i] = input[i] ^ key;
    fp = fopen("encode.txt", "w+");
    fprintf(fp, "\\");
    for (i = 0; i<len; i++)
    {
        fprintf(fp, "\\x%0.2x", output[i]);
        if ((i + 1) % 16 == 0)
            fprintf(fp, "\\n\\n");
    }
    fprintf(fp, "\\");
    fclose(fp);
    printf("dump the encoded shellcode to encode.txt OK!\\n");
    free(output);
}

int main()
{
    char sc[] =
        "\\x33\\xDB\\x53\\x68\\x72\\x6C\\x64\\x20\\x68\\x6F\\x20\\x77\\x6F\\x68\\x68\\x65\\x6C\\x6C\\x8B\\xC4\\x53\\x50\\x50\\x53\\xB8\\xEA\\x07\\xD5\\x77\\xFF\\xD0\\x90";
    encoder(sc, 0x44);
    getchar();
    return 0;
}
```