

# 攻防世界 逆向 hackme

原创

与日平肩以头抢地 于 2020-02-20 09:40:27 发布 收藏 536

分类专栏：逆向 文章标签：安全

版权声明：本文为博主原创文章，遵循CC 4.0 BY-SA 版权协议，转载请附上原文出处链接和本声明。

本文链接：[https://blog.csdn.net/fool\\_best/article/details/104405333](https://blog.csdn.net/fool_best/article/details/104405333)

版权



[逆向 专栏收录该内容](#)

12 篇文章 0 订阅

订阅专栏

## 攻防世界 逆向 hackme

(原创)

原题如下：

hackme 1 最佳Writeup由admin提供

难度系数： ★★2.0

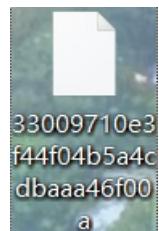
题目来源： XCTF 3rd-GCTF-2017

题目描述：暂无

题目场景：暂无

题目附件： 附件1 [https://blog.csdn.net/fool\\_best](https://blog.csdn.net/fool_best)

下载文件



IDA打开

IDA - 33009710e3f44f04b5a4cdbaaa46f00a C:\Users\dell\Desktop\33009710e3f44f04b5a4cdbaaa46f00a

文件(F) 编辑(E) 跳转(J) 搜索(H) 视图(V) 调试器 选项(O) 窗口(W) 帮助

库函数 常规函数 命令 数据 未知 外部符号

函数窗口

函数名称 段

- [sub\\_484D50](#) .t
- [sub\\_484D80](#) .t
- [sub\\_484D90](#) .t
- [sub\\_484DA0](#) .t
- [nullsub\\_4](#) .t
- [sub\\_484FC0](#) .t
- [sub\\_485020](#) .t
- [sub\\_485100](#) .t
- [sub\\_4851C0](#) .t
- [sub\\_4852C0](#) .t
- [sub\\_485360](#) .t
- [.cnnconv](#) .t

IDA View-A 十六进制视图-1 构体 枚举

```
.plt:00000000004002AB jmp $+5
=plt:00000000004002B0
=plt:00000000004002B0 ; ===== S U B R O U T I N E =====
=plt:00000000004002B0
=plt:00000000004002B0 ; Attributes: thunk
=plt:00000000004002B0
=plt:00000000004002B0
=plt:00000000004002B0 sub_4002B0 proc near ; CODE XREF: .plt:00000000004002AB+j
=plt:00000000004002B0
=plt:00000000004002B0
=plt:00000000004002B0 jmp cs:off_6B4218 ; sub_401800+F4+jp ...
=plt:00000000004002B0
=plt:00000000004002B0 endp
=plt:00000000004002B0
=plt:00000000004002B0 ; ===== S U B R O U T I N E =====
=plt:00000000004002B0
```

```

sub_4850300
sub_4854A0
sub_485550
sub_4855C0
sub_485630
sub_4856A0
sub_485860
sub_485980
sub_485B10
sub_485FA0
sub_486040
sub_4861B0
sub_4862F0
sub_486AA0
sub_486C70
sub_486C90
sub_486DC0
sub_486E00
sub_487000
sub_487070
sub_487430
sub_487530
sub_4875B0
sub_487630
sub_487710

行 22/735
输出窗口
400740: positive sp value has been found
Command "JumpEnter" failed
Python
AU: idle 向下 磁盘: 12GB

```

确定sub\_400F8E()函数找flag。

**函数窗口**

函数名称	段
sub_400A03	.t
sub_400AC6	.t
sub_400C00	.t
sub_400C10	.t
start	.t
sub_400EA0	.t
sub_400EE0	.t
sub_400F20	.t
sub_400F50	.t
<b>sub_400F8E</b>	<b>.t</b>
sub_4010F0	.t
sub_4013A0	.t
sub_401450	.t
sub_401490	.t
sub_4016B0	.t
sub_401720	.t
sub_4017C0	.t
sub_401800	.t
sub_401940	.t
sub_401980	.t
sub_401990	.t
sub_401B40	.t

**IDA View-A**

```

1 int64 sub_400F8E()
2 {
3     char v1[136]; // [rsp+10h] [rbp-B0h]
4     int v2; // [rsp+98h] [rbp-28h]
5     char v3; // [rsp+9Fh] [rbp-21h]
6     int v4; // [rsp+A0h] [rbp-20h]
7     unsigned __int8 v5; // [rsp+A6h] [rbp-1Ah]
8     char v6; // [rsp+A7h] [rbp-19h]
9     int v7; // [rsp+A8h] [rbp-18h]
10    int v8; // [rsp+ACh] [rbp-14h]
11    int v9; // [rsp+B0h] [rbp-10h]
12    int v10; // [rsp+B4h] [rbp-Ch]
13    _BOOL4 v11; // [rsp+B8h] [rbp-8h]
14    int i; // [rsp+BCh] [rbp-4h]
15
16    sub_407470((unsigned __int64)"Give me the password: ");
17    sub_4075A0((unsigned __int64)%s");
18    for ( i = 0; v1[i]; ++i )
19    ;
20    v11 = i == 22;
21    v10 = 10;
22    do
23    {
24        v7 = (signed int)sub_406D90("%s", v1) % 22;
25        v9 = 0;
26        v6 = byte_6B4270[v7];
27        v5 = v1[v7];
28        v4 = v7 + 1;
29        v8 = 0;
30        while ( v8 < v4 )
31        {
32            ++v8;
33            v9 = 1828812941 * v9 + 12345;
34        }
35        v3 = v9 ^ v5;
36        if ( v6 != ((unsigned __int8)v9 ^ v5) )

```

**图表概览**

输出窗口

```
4075A0: using guessed type __int64 __fastcall sub_4075A0(char);
```

伪代码如下：

```

__int64 sub_400F8E()
{
    char v1[136]; // [rsp+10h] [rbp-B0h]
    int v2; // [rsp+98h] [rbp-28h]
    char v3; // [rsp+9Fh] [rbp-21h]
    int v4; // [rsp+A0h] [rbp-20h]
    unsigned __int8 v5; // [rsp+A6h] [rbp-1Ah]
    char v6; // [rsp+A7h] [rbp-19h]
    int v7; // [rsp+A8h] [rbp-18h]
    int v8; // [rsp+ACh] [rbp-14h]
    int v9; // [rsp+B0h] [rbp-10h]
    int v10; // [rsp+B4h] [rbp-Ch]
    _BOOL4 v11; // [rsp+B8h] [rbp-8h]
    int i; // [rsp+BCh] [rbp-4h]

    sub_407470((unsigned __int64)"Give me the password: ");
    sub_4075A0((unsigned __int64)%s");
    for ( i = 0; v1[i]; ++i )
        ;
    v11 = i == 22;
    v10 = 10;
    do
    {
        v7 = (signed int)sub_406D90("%s", v1) % 22;
        v9 = 0;
        v6 = byte_6B4270[v7];
        v5 = v1[v7];
        v4 = v7 + 1;
        v8 = 0;
        while ( v8 < v4 )
        {
            ++v8;
            v9 = 1828812941 * v9 + 12345;
        }
        v3 = v9 ^ v5;
        if ( v6 != ((unsigned __int8)v9 ^ v5) )
            v11 = 0;
        --v10;
    }
    while ( v10 );
    if ( v11 )
        v2 = sub_407470((unsigned __int64)"Congras\n");
    else
        v2 = sub_407470((unsigned __int64)"Oh no!\n");
    return 0LL;
}

```

程序：验证输入22位->生成随机数验证10位数

用脚本运行

```
bs=[0x5F, 0xF2, 0x5E, 0x8B, 0x4E, 0x0E, 0xA3, 0xAA, 0xC7, 0x93,
    0x81, 0x3D, 0x5F, 0x74, 0xA3, 0x09, 0x91, 0x2B, 0x49, 0x28,
    0x93, 0x67]
flag=[0 for i in range(22)]
for index in range(22):
    b=bs[index]
    temp=0
    for i in range(index+1):
        temp=(0x6D01788D * temp + 0x3039)
    flag[index]=(temp^b)&0xff
print(''.join(map(chr,flag)))
```

得到flag{d826e6926098ef46}