

xctf攻防世界logmein writeup

原创

qq_112419837



于 2020-11-06 09:37:30 发布



125



收藏

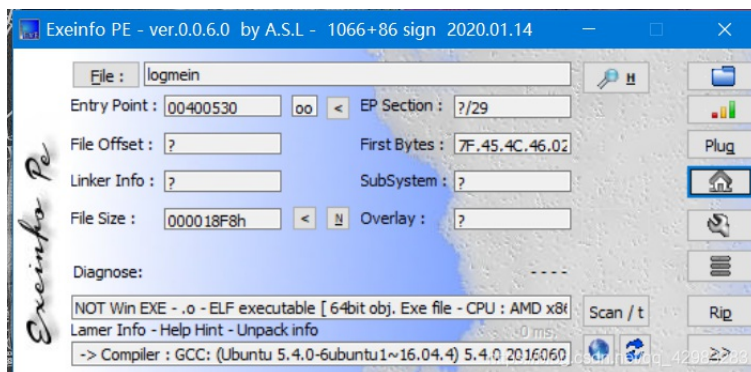
版权声明：本文为博主原创文章，遵循 [CC 4.0 BY-SA](#) 版权协议，转载请附上原文出处链接和本声明。

本文链接：https://blog.csdn.net/qq_42983283/article/details/109524860

版权

The screenshot shows a challenge card for 'logmein'. It features a title 'logmein' with a thumbs-up icon and the number '35'. Below the title, it says '最佳Writeup由Sec_Evil • Sec_evil提供'. The difficulty coefficient is '3.0' with three stars. The source is 'RC3 CTF 2016'. The description is '菜鸡开始接触一些基本的算法逆向了'. The scenario is '暂无'. There is a button for '附件1'. The URL 'https://blog.csdn.net/qq_42983283' is visible at the bottom right.

下载查看：



ida64打开，f5：

大意是输入个s，然后和v8比较一下长度，小了就要执行4007c0（退出）。

然后就是循环判断，s的长度还不能大于v8，那就是一样呗；接着拿s中的字符和一个算式比较，这个算式应该就是flag。

```

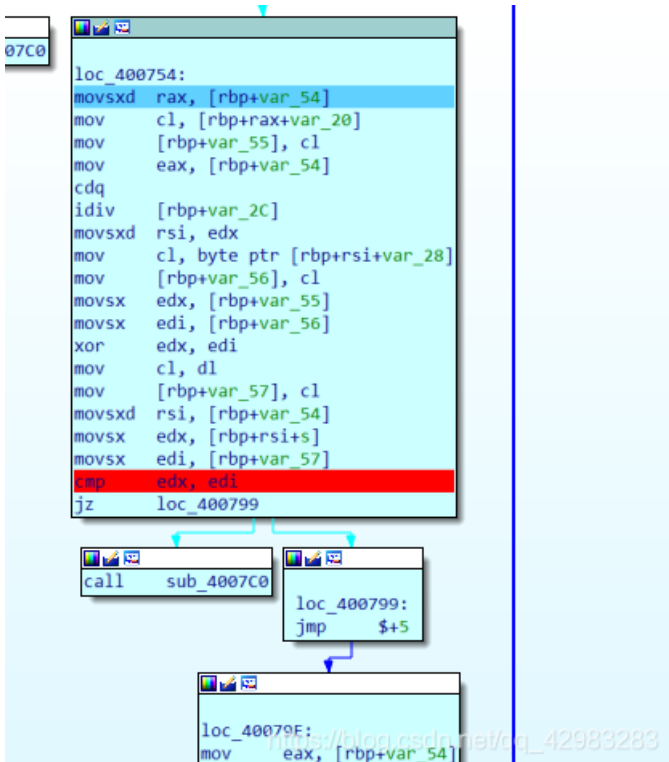
void __fastcall __noreturn main(__int64 a1, char **a2, char **a3)
{
    size_t v3; // rsi
    int i; // [rsp+3Ch] [rbp-54h]
    char s[36]; // [rsp+40h] [rbp-50h]
    int v6; // [rsp+64h] [rbp-2Ch]
    __int64 v7; // [rsp+68h] [rbp-28h]
    char v8[8]; // [rsp+70h] [rbp-20h]
    int v9; // [rsp+8Ch] [rbp-4h]

    v9 = 0;
    strcpy(v8, ":\\"AL_RT^L*.?+6/46");
    v7 = 28537194573619560LL;
    v6 = 7;
    printf("Welcome to the RC3 secure password guesser.\n", a2, a3);
    printf("To continue, you must enter the correct password.\n");
    printf("Enter your guess: ");
    __isoc99_scanf("%32s", s);
    v3 = strlen(s);
    if ( v3 < strlen(v8) )
        sub_4007C0(v8);
    for ( i = 0; i < strlen(s); ++i )
    {
        if ( i >= strlen(v8) )
            ((void (*)(void))sub_4007C0)();
        if ( s[i] != (char)*((__BYTE *)&v7 + i % v6) ^ v8[i] ) #这句关键, 就是flag
            ((void (*)(void))sub_4007C0)();
    }
    sub_4007F0();
}

```

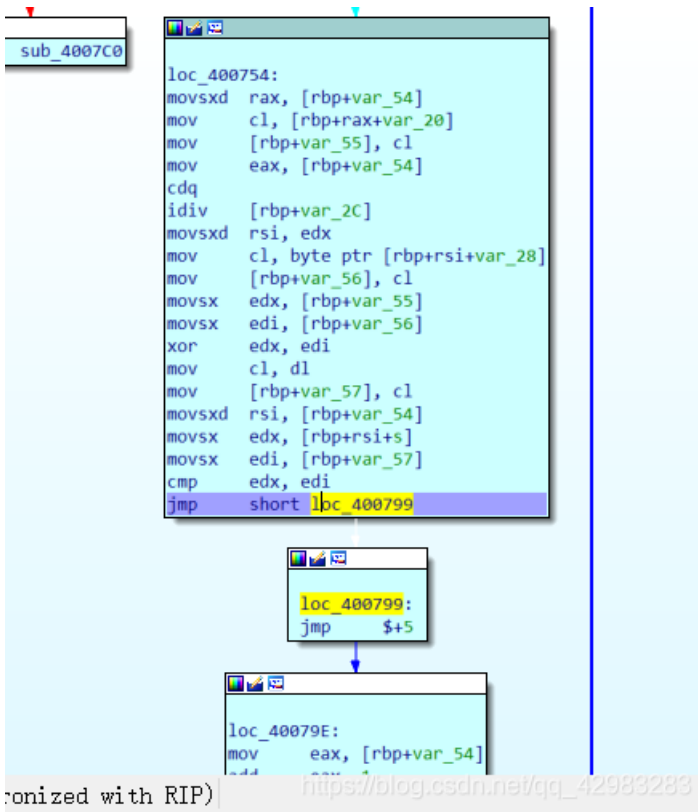
在第二个if处下断，执行几次就行了：

断下之后，在关键判断处下断，查看edx和edi的值：



因为我输入的全都是1，查看edi的值是52，edx的值是31，那么52就是第一个字符了。

我这里修改了cmp下面的跳转：



现在只设置一个断点，就是jmp那里，每次f9以后，查看edi的值就行了，如下：

5243332D323031362D584F524953475544



还有一种方法，可以直接看，那就是修改完跳转以后，注意cmp上面给edi赋值的那句movsx edi, [rbp+var_57]。

```
xor    edx, edi
mov    cl, dl
mov    [rbp+var_57], cl
movsxd rsi, [rbp+var_54]
movsx  edx, [rbp+rsi+s]
movsx  edi, [rbp+var_57]
cmp    edx, edi
jmp    short loc_400799
```

```
loc_400799:
jmp    $+5
```

双击就到了存储flag的地址:

```
889 db 52h ; R
88A db 68h ; h
88B db 3Ah ; :
88C db 0
88D db 0
88E db 0
88F db 0
890 db 31h ; 1
891 db 31h ; 1
892 db 31h ; 1
893 db 31h ; 1
894 db 31h ; 1
895 db 31h ; 1
896 db 31h ; 1
897 db 31h ; 1
```

一直f9就行了。

还可以编写c语言，直接输出:

```
#include <stdio.h>
void main()
{

    char v4[18]; // ST39_1

    int i; // [rsp+3Ch] [rbp-54h]

    int v8; // [rsp+64h] [rbp-2Ch]
    long long int v9; // [rsp+68h] [rbp-28h]
    char v10[18]; // [rsp+70h] [rbp-20h]

    strcpy(v10, ":\\"AL_RT^L*.*+6/46");
    v9 = 28537194573619560LL;
    v8 = 7;

    for ( i = 0; i < 17; ++i )
    {

        v4[i] = *((char *)&v9 + i % v8) ^ v10[i];

    }

    printf("%s",v4);
}
```