

xctf Mysterious

原创

菜逼的ctf之路 于 2020-10-14 20:38:22 发布 131 收藏 1

版权声明：本文为博主原创文章，遵循 [CC 4.0 BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) 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/weixin_45701079/article/details/109082801

版权

xctf Mysterious

一道水题，记录一下，毕竟难题也不会

看代码找函数，其他都没用，只看这个主要的代码

```
ExitProcess(0);
v10 = atoi(&String) + 1;
if ( v10 == 123 && v12 == 120 && v14 == 122 && v13 == 121 )
{
    strcpy(Text, "flag");
    memset(&v7, 0, 0xFCu);
    v8 = 0;
    v9 = 0;
    _itoa(v10, &v5, 10);
    strcat(Text, "{");
    strcat(Text, &v5);
    strcat(Text, "_");
    strcat(Text, "Buff3r_0v3rf|0w");
    strcat(Text, "}");
    MessageBoxA(0, Text, "well done", 0);
}
GetTimer(hwnd, 1, 0x250, TimerFunc);
```

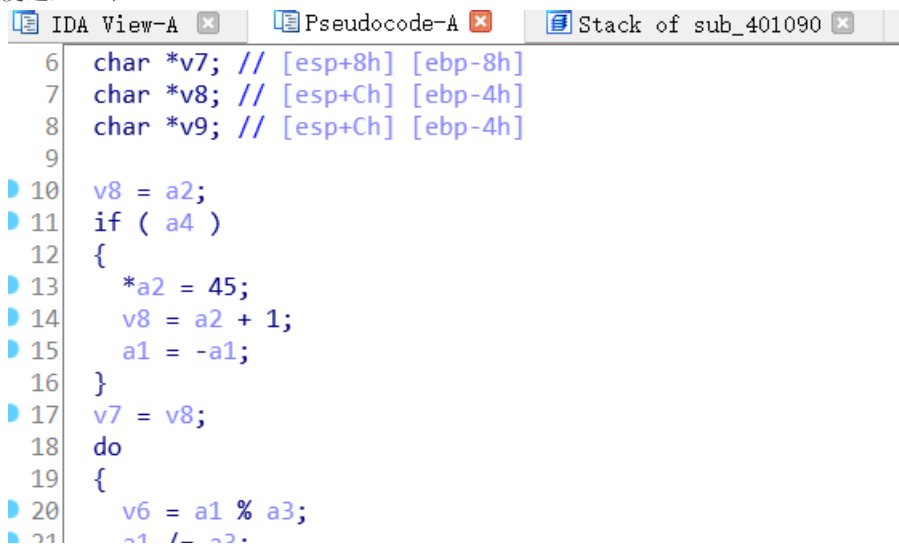
https://blog.csdn.net/weixin_45701079

这个很简单，通过逻辑，只要找到v5就行了，v5应该在_itoa()函数，

查看函数

```
char *__cdecl _itoa(int a1, char *a2, int a3)
{
    if ( a3 != 10 || a1 >= 0 )
        xtoa(a1, a2, a3, 0);
    else
        xtoa(a1, a2, 0xAu, 1);
    return a2;
}
```

两个函数调用一样，随便进入一个



The screenshot shows the IDA Pro interface with the assembly code for the _itoa function. The code is as follows:

```
6 char *v7; // [esp+8h] [ebp-8h]
7 char *v8; // [esp+Ch] [ebp-4h]
8 char *v9; // [esp+Ch] [ebp-4h]
9
10 v8 = a2;
11 if ( a4 )
12 {
13     *a2 = 45;
14     v8 = a2 + 1;
15     a1 = -a1;
16 }
17 v7 = v8;
18 do
19 {
20     v6 = a1 % a3;
21     a1 /= a3;
```

```

21     a1 /= a3,
22     if ( v6 <= 9 )
23         *v8 = v6 + 48;
24     else
25         *v8 = v6 + 87;
26     ++v8;
27 }
28 while ( a1 );
29 *v8 = 0;
30 v9 = v8 - 1;
31 do
32 {
33     v4 = *v9;
34     *v9 = *v7;
35     *v7 = v4;
36     --v9;
37     result = (int)(v7++ + 1);
38 }
39 while ( v7 < v9 );
40 return result;
41 }

```

https://blog.csdn.net/weixin_45701079

查看代码逻辑，主要核心代码在19-27，其他的都是赋值，这个循环的主要作用就是把数字转换为字符串，然后就懂了，v5就是字符串:123，然后加上去就可以了。