

# XCTF 新手 RE maze

原创

[A\\_dmins](#) 于 2019-07-03 17:35:02 发布 1205 收藏 4

分类专栏: [CTF题](#) [XCTF](#) [一天一道CTF](#)

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\\_42967398/article/details/94576708](https://blog.csdn.net/qq_42967398/article/details/94576708)

版权



[CTF题](#) 同时被 3 个专栏收录

115 篇文章 11 订阅

订阅专栏



[XCTF](#)

24 篇文章 0 订阅

订阅专栏



[一天一道CTF](#)

52 篇文章 5 订阅

订阅专栏

## XCTF 新手 RE maze

一天一道CTF题目, 能多不能少

今天没有心思复习，于是打开XCTF想找个题目做一下，结果发现这道题目有50多天没解决，，，想看看到底是什么题目，这么久没写，啊哈哈哈



The screenshot shows a CTF problem interface. At the top left is a '返回' (Return) button with a left arrow. To its right is a star icon and a timer showing '本题用时: 55天20时36分42秒'. The problem title is 'maze'. Below the title, the difficulty is '难度系数: ★★★★★ 4.0'. The source is '题目来源: NJUPT CTF 2017'. The description is '题目描述: 菜鸡想要走出菜狗设计的迷宫'. The scene is '题目场景: 暂无'. There is one attachment: '题目附件: 附件0'. A URL 'https://blog.csdn.net/qq\_42967398' is visible at the bottom right.

由题目得知这个题可能是迷宫，，，，，

下载文件，用ida打开（64位）

找到主函数~，太长了不好截图，就把源码贴出来吧，源码如下：

```
__int64 __fastcall main(__int64 a1, char **a2, char **a3)
{
    const char *v3; // rsi
    signed __int64 v4; // rbx
    signed int v5; // eax
    char v6; // bp
    char v7; // al
    const char *v8; // rdi
    __int64 v10; // [rsp+0h] [rbp-28h]

    v10 = 0LL;
    puts("Input flag:");
    scanf("%s", &s1, 0LL);
    if ( strlen(&s1) != 24 || (v3 = "nctf{", strcmp(&s1, "nctf{", 5uLL)) || *(&byte_6010BF + 24) != '}' ) )
    {
        LABEL_22:
        puts("Wrong flag!");
        exit(-1);
    }
    v4 = 5LL;
    if ( strlen(&s1) - 1 > 5 )
    {
        while ( 1 )
        {
            v5 = *(&s1 + v4);
            v6 = 0;
            if ( v5 > 'N' )
            {
```

```

v5 = (unsigned __int8)v5;
if ( (unsigned __int8)v5 == '0' )
{
    v7 = sub_400650((_DWORD *)&v10 + 1);
    goto LABEL_14;
}
if ( v5 == 'o' )
{
    v7 = sub_400660((int *)&v10 + 1);
    goto LABEL_14;
}
}
else
{
    v5 = (unsigned __int8)v5;
    if ( (unsigned __int8)v5 == '.' )
    {
        v7 = sub_400670(&v10, v3);
        goto LABEL_14;
    }
    if ( v5 == '0' )
    {
        v7 = sub_400680((int *)&v10);
LABEL_14:
        v6 = v7;
        goto LABEL_15;
    }
}
LABEL_15:
v3 = (const char *)HIDWORD(v10);
if ( !(unsigned __int8)sub_400690(asc_601060, HIDWORD(v10), (unsigned int)v10) )
    goto LABEL_22;
if ( ++v4 >= strlen(&s1) - 1 )
{
    if ( v6 )
        break;
LABEL_20:
    v8 = "Wrong flag!";
    goto LABEL_21;
}
}
if ( asc_601060[8 * (signed int)v10 + SHIDWORD(v10)] != '#' )
    goto LABEL_20;
v8 = "Congratulations!";
LABEL_21:
puts(v8);
return 0LL;
}

```

分析一波~~

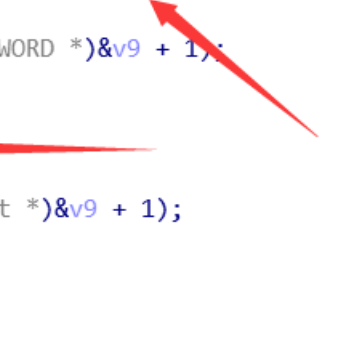
首先是输入为24位长的字符串，并且包括nctf{}，否则错误

所以就只剩下18位不知道，估摸着是路径

```
scanf("%s", &s1, 0LL);
if ( strlen(&s1) != 24 || strncmp(&s1, "nctf{", 5uLL) || *(&byte_6010BF + 24) != '}' )
{
ABEL_22:
    puts("Wrong flag!");
    exit(-1);
}
```

接下来发现有类似于四个方位的东西: o00.

```
{
    v4 = (unsigned __int8)v4;
    if ( (unsigned __int8)v4 == '0' )
    {
        v6 = sub_400650((_DWORD *)&v9 + 1);
        goto LABEL_14;
    }
    if ( v4 == 'o' )
    {
        v6 = sub_400660((int *)&v9 + 1);
        goto LABEL_14;
    }
}
else
{
    |
    v4 = (unsigned __int8)v4;
    if ( (unsigned __int8)v4 == '.' )
    {
        v6 = sub_400670(&v9);
        goto LABEL_14;
    }
    if ( v4 == '0' )
    {
        v6 = sub_400680((int *)&v9);
LABEL_14:
        v5 = v6;
        goto LABEL_15;
    }
}
}
```



[https://blog.csdn.net/qq\\_42967398](https://blog.csdn.net/qq_42967398)

容易知道v9就是一个二维数组

那么就很容易判断这四个分别代表什么方向了

查看这四个函数:

```

//0并且&v9 + 1
bool __fastcall sub_400650(_DWORD *a1)
{
    int v1; // eax
    v1 = (*a1)--;
    return v1 > 0;
}

//o并且&v9 + 1
bool __fastcall sub_400660(int *a1)
{
    int v1; // eax
    v1 = *a1 + 1;
    *a1 = v1;
    return v1 < 8;
}

//.并且&v9
bool __fastcall sub_400670(_DWORD *a1)
{
    int v1; // eax
    v1 = (*a1)--;
    return v1 > 0;
}

//0并且&v9
bool __fastcall sub_400680(int *a1)
{
    int v1; // eax
    v1 = *a1 + 1;
    *a1 = v1;
    return v1 < 8;
}

```

得知了这些信息，那么我们就很容易知道方向了  
v9+1表示二维的，也就是左右，  
所以这四个表示方向分别为：

```

0左
o右
.上
0下

```

分析完之后，还有一个信息就是，这个地图是8\*8的  
 并且还有个地图好像还不知道，继续往下看，，，  
 看到有Congratulations!  
 并且知道是要走到 # 位置，否则就会跳到Wrong flag!

```

LABEL_15:
    if ( !(unsigned __int8)sub_400690(asc_601060, HIDWORD(v9), (unsigned int)v9) )
        goto LABEL_22;
    if ( ++v3 >= strlen(&s1) - 1 )
    {
        if ( v5 )
            break;
    }
LABEL_20:
    v7 = "Wrong flag!";
    goto LABEL_21;
}
}
}
if ( asc_601060[8 * (signed int)v9 + SHIDWORD(v9)] != '#' )
    goto LABEL_20;
v7 = "Congratulations!";

```

[https://blog.csdn.net/qq\\_42967398](https://blog.csdn.net/qq_42967398)

双击asc\_601060查看，得到地图：

```

160 asc_601060      db ' ***** * **** * **** * *** *# *** *** *** *****',0

```

不过这样不好看，用01来进行表示：

```

s = " ***** * **** * **** * *** *# *** *** *** *****"
x = ""
for i in s:
    if i == ' ':
        x += '0'
    elif i == '*':
        x += '1'
    else:
        x += i
print(x)

00111111
10001001
11101011
11001011
1001#001
11011101
11000001
11111111

```

最后从头开始走到#就可以了，只能走0

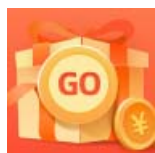
最后得到flag: `nctf{o0o0000000oooo..00}`

50多天的题目终于得到解决了~~:

恭喜您答对了

难度 ★★★★★ 耗时: 55天20时55分20秒 积分: 3.66 金币: 4+2

以前还是学识太浅，感觉动都动不了



[创作打卡挑战赛](#)

[赢取流量/现金/CSDN周边激励大奖](#)