2019第三届强网杯-强网先锋-ADwp

原创

n0paben 于 2019-05-27 11:29:53 发布 4302 收藏 5
分类专栏： CTF 文章标签： ctf第三届强网杯

周末参加了一下强网杯，emmm这些题目不是我等弟弟所能解答的。

鲲or鳗orGame题目wp https://blog.csdn.net/or1d1/article/details/90599659

其他大佬的wp

https://skysec.top/2019/05/25/2019-%E5%BC%BA%E7%BD%91%E6%9D%AFonline-Web-Writeup/#upload

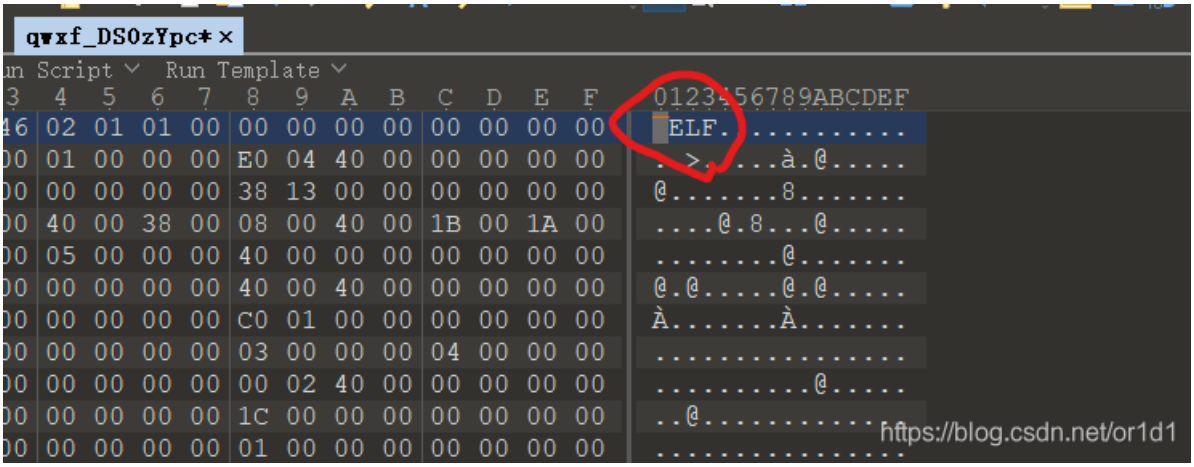
https://mp.weixin.qq.com/s?__biz=MzIxMDYyNTk3Nw==&mid=2247484435&idx=1&sn=8c8e3cf479f56b3ab324ae6f4774ef48&chksm=976

https://www.zhaoj.in/read-5873.html?tdsourcetag=s_pctim_aiomsg

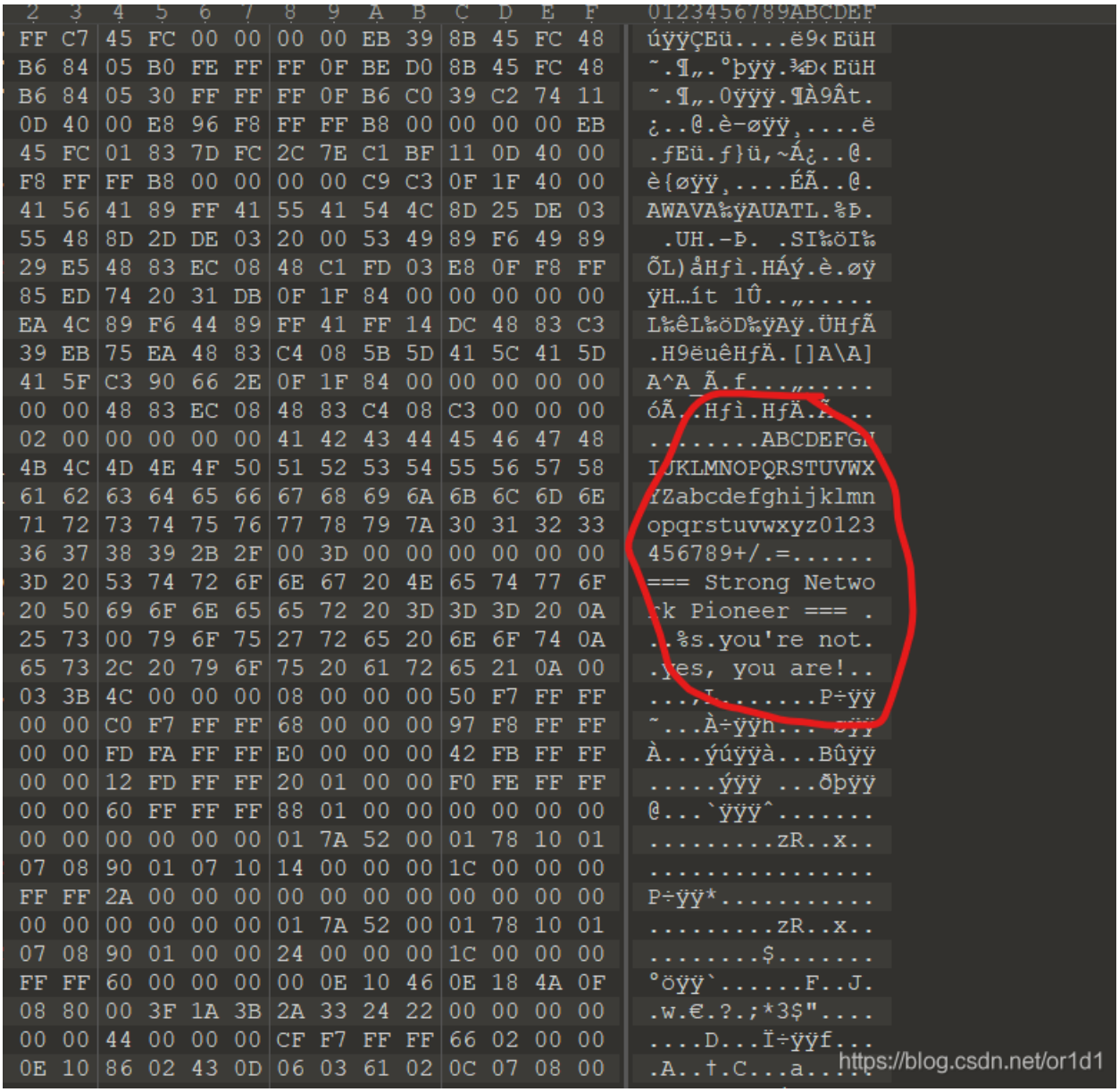
---

强网先锋-ADwp：

这道题下载下来是一个没有后缀的文件，emm怎么搞，先010editor试一下

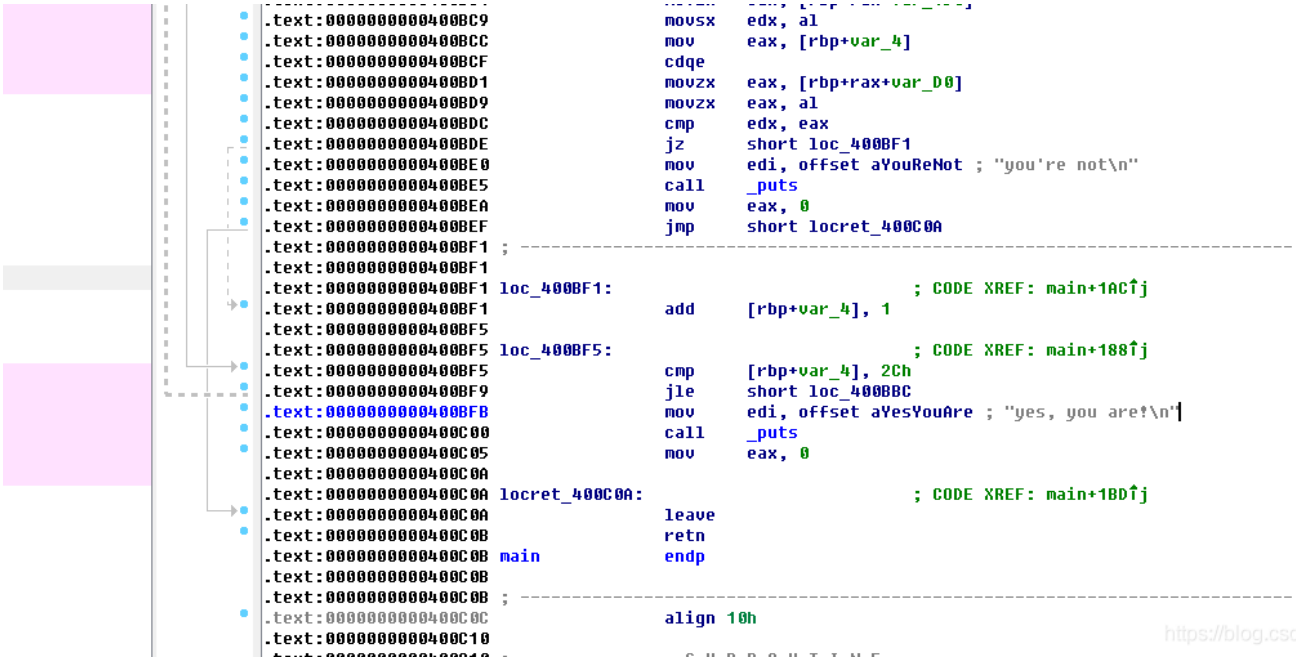


发现有个ELF，然后往下看



发现有 you are not 的字样，暂时是没有类似 flag 的内容，既然是 elf 文件，那么就丢进 kali 用 elfread 打开一下，发现还是没有什么有用的信息。

噢噢 他是一道逆向题（我。。。）丢进ida看一下



在 main 函数里看到了you are not 的字样，f5 试试

```
52  v43 = 7956005061827062375LL;
53  v44 = 105;
54  v5 = 90;
55  v6 = 109;
56  v7 = 120;
57  v8 = 104;
58  v9 = 90;
59  v10 = 51;
60  v11 = 116;
61  v12 = 116;
62  v13 = 89;
63  v14 = 87;
64  v15 = 90;
65  v16 = 104;
66  v17 = 97;
67  v18 = 51;
68  v19 = 86;
69  v20 = 104;
70  v21 = 97;
71  v22 = 87;
72  v23 = 120;
73  v24 = 104;
74  v25 = 97;
75  v26 = 88;
76  v27 = 70;
77  v28 = 112;
78  v29 = 89;
79  v30 = 87;
80  v31 = 53;
81  v32 = 107;
82  v33 = 89;
83  v34 = 87;
84  v35 = 57;
85  v36 = 105;
86  v37 = 102;
87  v38 = 81;
88  v39 = 61;
89  v40 = 61;
90  v46 = &v45;
91  sub_4005B7(&v45, (__int64)v4);
92  for ( i = 0; i <= 44; ++i )
93  {
94    if ( v4[i] != (unsigned __int8)*(&v5 + i) )
95    {
96      puts("you're not\n");
97      return 0LL;

00000AF4 main:67
```


emmm 这是什么鬼，for 循环，看一下上面的数字，好像有点熟悉，转成字符试一下

Functions window

Function name
- _init_proc
- **_puts**
- **_strlen**
- **_strchr**
- **__libc_start_main**
- __isoc99_scanf
- __gmon_start__
- start
- sub_400510
- sub_400540
- sub_400580
- sub_4005B0
- sub_4005B7
- sub_40081D
- main
- init
- fini
- _term_proc
- **puts**
- **strlen**
- **strchr**
- **__libc_start_main**
- __isoc99_scanf

IDA View-A | Pseudocode-A | Stack of main | Hex View-1 | Structures | Enums

```
49  __isoc99_scanf("%s", &v45);
50  v41 = 7953769703030221169LL;
51  v42 = 7954876941086586983LL;
52  v43 = 7956005061827062375LL;
53  v44 = 'i';
54  v5 = '2';
55  v6 = 'm';
56  v7 = 'x';
57  v8 = 'h';
58  v9 = '2';
59  v10 = '3';
60  v11 = 't';
61  v12 = 't';
62  v13 = 'Y';
63  v14 = 'W';
64  v15 = 'Z';
65  v16 = 'h';
66  v17 = 'a';
67  v18 = '3';
68  v19 = 'V';
69  v20 = 'h';
70  v21 = 'a';
71  v22 = 'W';
72  v23 = 'x';
73  v24 = 'h';
74  v25 = 'a';
75  v26 = 'X';
76  v27 = 'F';
77  v28 = 'p';
78  v29 = 'Y';
79  v30 = 'W';
80  v31 = '5';
81  v32 = 'k';
82  v33 = 'Y';
83  v34 = 'W';
84  v35 = '9';
85  v36 = 'i';
86  v37 = 'f';
87  v38 = 'Q';
88  v39 = '=';
89  v40 = '=';
90  v46 = &v45;
91  sub_4005B7(&v45, (__int64)v4);
92  for ( i = 0; i <= 44; ++i )
```

吆西，后面还有两个等于号，base64 先试一下

```
iZmxhZ3ttYWZha3VhaWxhaXFpYW5kYW9ifQ==
```

发现解不出来，woc，弃赛，告辞！

又看了一下上面，v44是 int 型别的都是 char 型，去掉44试一下

```
__int64 __fastcall main(__int64 a1, char **a2, char **a3)
{
  char v4[128]; // [sp+0h] [bp-150h]@1
  char v5; // [sp+80h] [bp-D0h]@1
  char v6; // [sp+81h] [bp-CFh]@1
  char v7; // [sp+82h] [bp-CEh]@1
  char v8; // [sp+83h] [bp-CDh]@1
  char v9; // [sp+84h] [bp-CCh]@1
  char v10; // [sp+85h] [bp-CBh]@1
  char v11; // [sp+86h] [bp-CAh]@1
  char v12; // [sp+87h] [bp-C9h]@1
  char v13; // [sp+88h] [bp-C8h]@1
  char v14; // [sp+89h] [bp-C7h]@1
  char v15; // [sp+8Ah] [bp-C6h]@1
  char v16; // [sp+8Bh] [bp-C5h]@1
  char v17; // [sp+8Ch] [bp-C4h]@1
  char v18; // [sp+8Dh] [bp-C3h]@1
  char v19; // [sp+8Eh] [bp-C2h]@1
  char v20; // [sp+8Fh] [bp-C1h]@1
  char v21; // [sp+90h] [bp-C0h]@1
  char v22; // [sp+91h] [bp-BFh]@1
  char v23; // [sp+92h] [bp-BEh]@1
  char v24; // [sp+93h] [bp-BDh]@1
  char v25; // [sp+94h] [bp-BCh]@1
  char v26; // [sp+95h] [bp-BBh]@1
  char v27; // [sp+96h] [bp-BAh]@1
  char v28; // [sp+97h] [bp-B9h]@1
  char v29; // [sp+98h] [bp-B8h]@1
  char v30; // [sp+99h] [bp-B7h]@1
  char v31; // [sp+9Ah] [bp-B6h]@1
  char v32; // [sp+9Bh] [bp-B5h]@1
  char v33; // [sp+9Ch] [bp-B4h]@1
  char v34; // [sp+9Dh] [bp-B3h]@1
  char v35; // [sp+9Eh] [bp-B2h]@1
  char v36; // [sp+9Fh] [bp-B1h]@1
  char v37; // [sp+A0h] [bp-B0h]@1
  char v38; // [sp+A1h] [bp-AFh]@1
  char v39; // [sp+A2h] [bp-AEh]@1
  char v40; // [sp+A3h] [bp-ADh]@1
  __int64 v41; // [sp+B0h] [bp-A0h]@1
  __int64 v42; // [sp+B8h] [bp-98h]@1
  __int64 v43; // [sp+C0h] [bp-90h]@1
  __int16 v44; // [sp+C8h] [bp-88h]@1
  char v45; // [sp+D0h] [bp-80h]@1
  char *v46; // [sp+140h] [bp-10h]@1
  int i; // [sp+14Ch] [bp-4h]@1
```


yes，you are!!!!