

攻防世界 Reverse高手进阶区 2分题 elrond32

原创

思源湖的鱼 于 2020-12-13 14:28:33 发布 127 收藏

分类专栏: [ctf](#) 文章标签: [ctf](#) [攻防世界](#) [reverse](#)

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/weixin_44604541/article/details/111111593

版权

CTF

[ctf 专栏收录该内容](#)

200 篇文章 23 订阅

订阅专栏

前言

继续ctf的旅程

攻防世界Reverse高手进阶区的2分题

本篇是elrond32的writeup

发现攻防世界的题目分数是动态的

就仅以做题时的分数为准了

解题过程

PE查壳



扔进IDA

```
1 int __cdecl main(int a1, char **a2)
2 {
3     if ( a1 > 1 && sub_8048414(a2[1], 0) )
4     {
```

```

7 | }
8 | puts("Access granted");
9 | sub_8048538(a2[1]);
10 | }
11 | else
12 | {
13 |     puts("Access denied");
14 | }
15 | return 0;
16 | }
17 | }
18 | }
19 | }
20 | }
21 | }
22 | }
23 | }
24 | }
25 | }
26 | }
27 | }
28 | }
29 | }
30 | }
31 | }
32 | }
33 | }
34 | }
35 | }
36 | }
37 | }
38 | }
39 | }
40 | }
41 | }
42 | }
43 | }
44 | }
45 | }
46 | }
47 | }
48 | }
49 | }
50 | }
51 | }

```

```

1 | signed int __cdecl sub_8048414(_BYTE *a1, int a2)
2 | {
3 |     signed int result; // eax
4 |
5 |     switch ( a2 )
6 |     {
7 |     case 0:
8 |         if ( *a1 == 105 )
9 |             goto LABEL_19;
10 |         result = 0;
11 |         break;
12 |     case 1:
13 |         if ( *a1 == 101 )
14 |             goto LABEL_19;
15 |         result = 0;
16 |         break;
17 |     case 3:
18 |         if ( *a1 == 110 )
19 |             goto LABEL_19;
20 |         result = 0;
21 |         break;
22 |     case 4:
23 |         if ( *a1 == 100 )
24 |             goto LABEL_19;
25 |         result = 0;
26 |         break;
27 |     case 5:
28 |         if ( *a1 == 97 )
29 |             goto LABEL_19;
30 |         result = 0;
31 |         break;
32 |     case 6:
33 |         if ( *a1 == 103 )
34 |             goto LABEL_19;
35 |         result = 0;
36 |         break;
37 |     case 7:
38 |         if ( *a1 == 115 )
39 |             goto LABEL_19;
40 |         result = 0;
41 |         break;
42 |     case 9:
43 |         if ( *a1 == 114 )
44 |             goto LABEL_19;
45 |     LABEL_19:
46 |         result = sub_8048414(a1 + 1, 7 * (a2 + 1) % 11);
47 |         else
48 |             result = 0;
49 |         break;
50 |     default:
51 |         result = 1;
52 |         break;
53 |     }
54 | }
55 | }
56 | }
57 | }
58 | }
59 | }
60 | }
61 | }
62 | }
63 | }
64 | }
65 | }
66 | }
67 | }
68 | }
69 | }
70 | }
71 | }
72 | }
73 | }
74 | }
75 | }
76 | }
77 | }
78 | }
79 | }
80 | }
81 | }
82 | }
83 | }
84 | }
85 | }
86 | }
87 | }
88 | }
89 | }
90 | }
91 | }
92 | }
93 | }
94 | }
95 | }
96 | }
97 | }
98 | }
99 | }
100 | }

```

```

1 | int __cdecl sub_8048538(int a1)
2 | {
3 |     int v2[32]; // [esp+18h] [ebp-A0h]
4 |     int i; // [esp+9Ch] [ebp-1Ch]
5 |
6 |     qmemcpy(v2, &unk_8048760, sizeof(v2));
7 |     for ( i = 0; i <= 32; ++i )
8 |         putchar(v2[i] ^ *(char*)(a1 + i % 8));
9 |     return putchar(10);
10 | }

```

一个验证输入
一个负责输出

跟进 8048769, 这里需要 $33*4=132=16*8+1$ 个字节的内容, 即需要前往 hex 窗口中找 8 行加 1 个 4 字节的数据, 还要注意小端

序的问

题 (还好这里每个 32bit 的前 3 个字节都是 0x00)

```
08048760 0f 00 00 00 1f 00 00 00 04 00 00 00 09 00 00 00 .....
08048770 1c 00 00 00 12 00 00 00 42 00 00 00 09 00 00 00 .....B.....
08048780 0c 00 00 00 44 00 00 00 0d 00 00 00 07 00 00 00 ....D.....
08048790 09 00 00 00 06 00 00 00 2d 00 00 00 37 00 00 00 .....7...
080487a0 59 00 00 00 1e 00 00 00 00 00 00 00 59 00 00 00 Y.....Y...
080487b0 0f 00 00 00 08 00 00 00 1c 00 00 00 23 00 00 00 .....#...
080487c0 36 00 00 00 07 00 00 00 55 00 00 00 02 00 00 00 6.....U.....
080487d0 0c 00 00 00 08 00 00 00 41 00 00 00 0a 00 00 00 .....A.....
080487e0 14 00 00 00 41 63 63 65 73 73 20 67 72 61 6e 74 ....Access-grant
```

根据题意

得到脚本

```
a='ie ndags r'
x=0
s=[]
for i in range(8):
    x=7*x%11
    s.append(a[x])
    x+=1
print(''.join(s))
data=[15, 31, 4, 9, 28, 18, 66, 9, 12, 68, 13, 7, 9, 6, 45, 55, 89, 30, 0, 89, 15, 8, 28, 35, 54, 7, 85, 2, 12,
8, 65, 10, 20]
for i in range(33):
    print(chr(ord(s[i%8])^data[i]),end='')
```

得到flag: `flag{s0me7hing_s0me7hinG_t0lki3n}`

结语

逻辑比较清晰