

攻防世界 Reverse高手进阶区 2分题 simple-check-100

原创

思源湖的鱼 于 2020-12-09 13:56:45 发布 80 收藏

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

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/110920947

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simple-check-100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

```
int __cdecl main(int argc, const char **argv, const char **envp)
{
    void *v3; // esp
    void *v4; // esp
```

```
void v4; // esp
char *v6; // [esp+4h] [ebp-44h]
char v7; // [esp+8h] [ebp-40h]
char v8; // [esp+1Bh] [ebp-2Dh]
char *v9; // [esp+1Ch] [ebp-2Ch]
int v10; // [esp+20h] [ebp-28h]
char v11; // [esp+25h] [ebp-23h]
char v12; // [esp+26h] [ebp-22h]
char v13; // [esp+27h] [ebp-21h]
char v14; // [esp+28h] [ebp-20h]
char v15; // [esp+29h] [ebp-1Fh]
char v16; // [esp+2Ah] [ebp-1Eh]
char v17; // [esp+2Bh] [ebp-1Dh]
char v18; // [esp+2Ch] [ebp-1Ch]
char v19; // [esp+2Dh] [ebp-1Bh]
char v20; // [esp+2Eh] [ebp-1Ah]
char v21; // [esp+2Fh] [ebp-19h]
char v22; // [esp+30h] [ebp-18h]
char v23; // [esp+31h] [ebp-17h]
char v24; // [esp+32h] [ebp-16h]
char v25; // [esp+33h] [ebp-15h]
char v26; // [esp+34h] [ebp-14h]
char v27; // [esp+35h] [ebp-13h]
char v28; // [esp+36h] [ebp-12h]
char v29; // [esp+37h] [ebp-11h]
char v30; // [esp+38h] [ebp-10h]
char v31; // [esp+39h] [ebp-Fh]
char v32; // [esp+3Ah] [ebp-Eh]
char v33; // [esp+3Bh] [ebp-Dh]
char v34; // [esp+3Ch] [ebp-Ch]
char v35; // [esp+3Dh] [ebp-Bh]
char v36; // [esp+3Eh] [ebp-Ah]
char v37; // [esp+3Fh] [ebp-9h]
int *v38; // [esp+40h] [ebp-8h]
```

```
v38 = &argc;
```

```
__main();
```

```
v8 = 84;
```

```
v37 = -56;
```

```
v36 = 126;
```

```
v35 = -29;
```

```
v34 = 100;
```

```
v33 = -57;
```

```
v32 = 22;
```

```
v31 = -102;
```

```
v30 = -51;
```

```
v29 = 17;
```

```
v28 = 101;
```

```
v27 = 50;
```

```
v26 = 45;
```

```
v25 = -29;
```

```
v24 = -45;
```

```
v23 = 67;
```

```
v22 = -110;
```

```
v21 = -87;
```

```
v20 = -99;
```

```
v19 = -46;
```

```
v18 = -26;
```

```
v17 = 109;
```

```
v16 = 44;
```

```

v15 = -45;
v14 = -74;
v13 = -67;
v12 = -2;
v11 = 106;
v10 = 19;
v3 = alloca(32);
v4 = alloca(32);
v9 = &v7;
printf("Key: ");
v6 = v9;
scanf("%s", v9);
if ( check_key(v9) )
    interesting_function(&v8);
else
    puts("Wrong");
return 0;
}

```

```

1 bool __cdecl check_key(int a1)
2 {
3     signed int i; // [esp+8h] [ebp-8h]
4     int v3; // [esp+Ch] [ebp-4h]
5
6     v3 = 0;
7     for ( i = 0; i <= 4; ++i )
8         v3 += *(DWORD*)(4 * i + a1);
9     return v3 == -559038737;
10 }

```

```

1 int __cdecl interesting_function(int a1)
2 {
3     int *result; // eax
4     unsigned int v2; // [esp+1Ch] [ebp-1Ch]
5     int *v3; // [esp+20h] [ebp-18h]
6     int v4; // [esp+24h] [ebp-14h]
7     int j; // [esp+28h] [ebp-10h]
8     int i; // [esp+2Ch] [ebp-Ch]
9
10    result = (int *)a1;
11    v4 = a1;
12    for ( i = 0; i <= 6; ++i )
13    {
14        v2 = *(DWORD*)(4 * i + v4) ^ 0xDEADBEEF;
15        result = (int *)&v2;
16        v3 = (int *)&v2;
17        for ( j = 3; j >= 0; --j )
18            result = (int *)putchar((char)((_BYTE *)v3 + j) ^ flag_data[4 * i + j]);
19    }
20    return result;
21 }

```

https://blog.csdn.net/weixin_44604541

。。

我看那你是在为难我胖虎

用gdb动态调试

尝试了下windows下会给乱码

要在linux下调试

```
0032| 0x7fffffff080 --> 0x7fffffff1c8 --> 0x7fffffff4db ("/root/Desktop/xctf
wn/simple-check-100")
0040| 0x7fffffff088 --> 0x100000000
0048| 0x7fffffff090 --> 0x0
0056| 0x7fffffff098 --> 0xe37ec85400000000
[-----]
Legend: code, data, rodata, value
0x00000000004008e4 in main ()
gdb-peda$ c
Continuing.
flag_is_you_know_cracking!!![Inferior 1 (process 4232) exited normally]
14604541
```

得到flag

结语

动态调试