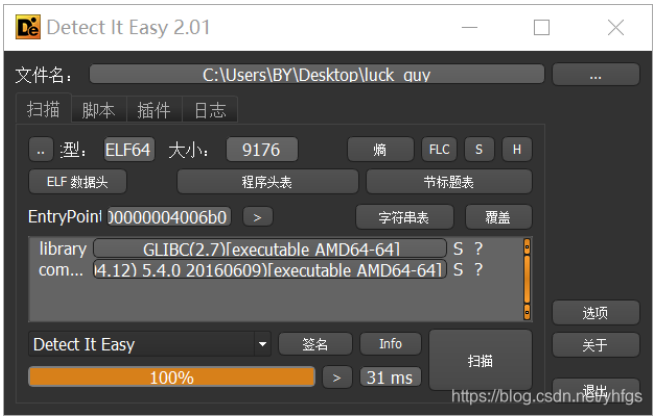
buuctf——luck_guy

原创

re3sry 于 2021-05-25 19:07:35 发布 161 收藏

1.下载得到文件luck_guy，查壳，丢到DIE中。

无壳，64位文件。



2.丢到IDA中，找到main函数，F5.分析代码，发现主要函数在patch_me。

```
int __cdecl main(int argc, const char **argv, const char **envp)
{
  unsigned int v4; // [rsp+14h] [rbp-Ch] BYREF
  unsigned __int64 v5; // [rsp+18h] [rbp-8h]

  v5 = __readfsqword(0x28u);
  welcome(argc, argv, envp);
  puts("_________________");
  puts("try to patch me and find flag");
  v4 = 0;
  puts("please input a lucky number");
  __isoc99_scanf("%d", &v4);
  patch_me(v4);
  puts("OK,see you again");
  return 0;
}
```


3.跟进patch_me函数。发现函数get_flag。直接跟进。

```
int __fastcall patch_me(int a1)
{
  int result; // eax

  if ( a1 % 2 == 1 )
    result = puts("just finished");
  else
    result = get_flag();
  return result;
}
```

```
case 1:
  puts("OK, it's flag:");
  memset(&s, 0, 0x28uLL);
  strcat((char *)&s, f1);                 //  (flag) s==GXY{do_not_
  strcat((char *)&s, &f2);                //  (flag) s==’GXY{do_not_‘+f2
  printf("%s", (const char *)&s);
  break;
case 2:
  printf("Solar not like you");
  break;
case 3:
  printf("Solar want a girlfriend");
  break;
case 4:
  s = 0x7F666F6067756369LL;
  v5 = 0;
  strcat(&f2, (const char *)&s);
  break;
case 5:
  for ( j = 0; j <= 7; ++j )
  {
    if ( j % 2 == 1 )
      *(&f2 + j) -= 2;
    else
      --*(&f2 + j);
```


4.发现flag为是s，而s为f1+f2，点击f1得到字符串 `f1 db 'GXY{do_not_',0 ; DATA XREF: get_flag+9E↑o`

现在就差f2，分析代码，猜测代码执行顺序可能是4-5-1，直接上脚本。注：（case 4中）s中的存储方式应该是小端序存储要逆序存储。

luck_guy.py - C:\Users\BY\Desktop\luck_guy1\luck_guy.py (3.9.2)

File Edit Format Run Options Window Help

```
s=''
for i in [0x69, 0x63, 0x75, 0x67, 0x60, 0x6F, 0x66, 0x7F]:
    s=s+chr(i)
flag='GXY{do_not_'
for i in range(7):
    if i%2==1:
        flag=flag+chr(ord(s[i])-2)
    else:
        flag=flag+chr(ord(s[i])-1)
print(flag)
```


5.get flag

GXY{do_not_hate_me}