# CTF|check writeup

原创

一个不融化的雪人 于 2020-06-20 16:24:51 发布 274 收藏



## CTF|check writeup

总体思路是想让你输入一串字符
输入的字符要使在400f51函数返回值为true 则到pwn环节

```
4  sub_400FC0();
5  sub_400A24();
6  if ( !(unsigned int)sub_400F51() )
7  {
8    puts("you play CTF like cxk~");
9    exit(0);
10 }
11 puts("OK let us pwn pwn pwn!\n");
```

函数里面条件要求几个变量等于某些值

```
1 _BOOL8 sub_400F51()
2 {
3   puts("there is no thing here,run as fast as you can~");
4   return byte_602060 == 90 && byte_602061 == 88 && byte_602062 == 90 && byte_602063 == 108;
5 }
```

一开始的想法是 这些变量都在bss段 想试试能不能往bss写入它要求的值

```
.bss:000000000060204C dword_60204C    dd ?
.bss:000000000060204C
.bss:0000000000602050                 align 20h
.bss:0000000000602060 byte_602060     db ?
.bss:0000000000602060
.bss:0000000000602061 byte_602061     db ?
.bss:0000000000602061
.bss:0000000000602062 byte_602062     db ?
.bss:0000000000602062
.bss:0000000000602063 byte_602063     db ?
.bss:0000000000602063
.bss:0000000000602064                 db    ? ;
.bss:0000000000602065                 db    ? ;
```



```
gdb-peda$ vmmap
Warning: not running
Start              End                Perm      Name
0x004006e0         0x0040122d         rx-p      /home/ubuntu16042/check
0x00400238         0x00401604         r--p      /home/ubuntu16042/check
0x00601db8         0x006020a8         rw-p      /home/ubuntu16042/check
gdb-peda$
```

Floppy Disk  Templates  Videos  1.ra

然后发现gbd也没办法调试
然后我就开始看 给出函数的内容 ...

我是从下往上看 条件中需要602060处值为90 所以v2应该等于25 因为这样对应v38 = 90
以此类推 v2=25；v3=23；v4=25；v5=37

```
199     }
200     byte_602060 = *(&v13 + v2);
201     byte_602061 = *(&v13 + v3);
202     byte_602062 = *(&v13 + v4);
203     byte_602063 = *(&v13 + v5);
204   }
205   return __readfsqword(0x28u) ^ v79;
206 }
```

希望通过这些值反推回去
然后看到函数中if（v7=1） if（v7=2）的分支都将v5最后赋值为64 不满足要求
然后我就确定突破口在v7=0的分支上 ， 而对于v7等于零意思是在这个三次的循环中没有出现v6为零的情况，即输入的字符串没有0.

```
for ( i = 0; i <= 2; ++i )
{
  if ( s[v6] )
  {
    v0 = v6++;
    *(&v10 + i) = s[v0];
  }
```

绕后有着上面不知道对还是错的思路 我就开始v10 v11 v12的值
v12 和0011 1111相与 得到 v5 我推出v12 = 37（和v5一样）
然后v12右移六位后为零 v4 = 4*（v11&0xF） 推出v11=6
然后再往上推 就觉得不对劲 估计思路错了
没能通过secret key这关…

```
v5 = v12 & 0x3F;
```

```
while ( v6 < v9 )
{
  v7 = 0;
  for ( i = 0; i <= 2; ++i )
  {
    if ( s[v6] )
    {
      v0 = v6++;
      *(&v10 + i) = s[v0];
    }
    else
    {
      *(&v10 + i) = 0;
      if ( i == 1 )
      {
        v7 = 1;
        break;
      }
      if ( i == 2 )
      {
        v7 = 2;
        break;
      }
    }
  }
  if ( !v7 )
  {
    v2 = v10 >> 2;
    v3 = 16 * (v10 & 3) + (v11 >> 4);
    v4 = 4 * (v11 & 0xF) + (v12 >> 6);
    v5 = v12 & 0x3F;
  }
  if ( v7 == 1 )
  {
    v2 = v10 >> 2;
    v3 = 16 * (v10 & 3) + (v11 >> 4);
    v4 = 64;
    v5 = 64;
  }
```



```
if ( v7 == 2 )
{
  v2 = v10 >> 2;
  v3 = 16 * (v10 & 3) + (v11 >> 4);
  v4 = 4 * (v11 & 0xF) + (v12 >> 6);
  v5 = 64;
}
```