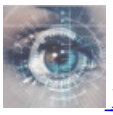
XCTF getit

原创

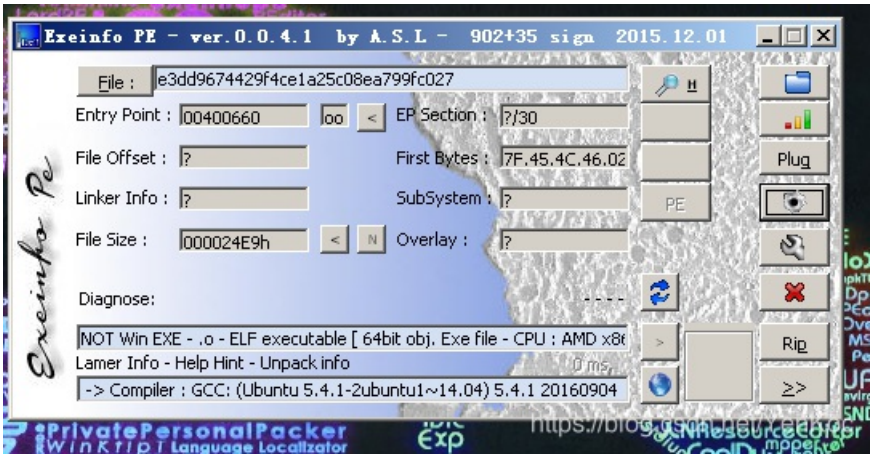
YenKoc 于 2020-01-15 17:01:01 发布 455 收藏

分类专栏： XCTF

本文链接：https://blog.csdn.net/YenKoc/article/details/103992641



一.查壳



是linux的文件。没加壳

二.拖入ida

```
1  int __cdecl main(int argc, const char **argv, const char **envp)
2  {
3    char v3; // al@4
4    int result; // eax@10
5    __int64 v5; // rbx@10
6    __int64 v6; // [sp+0h] [bp-40h]@0
7    int i; // [sp+4h] [bp-3Ch]@7
8    FILE *stream; // [sp+8h] [bp-38h]@7
9    char filename[8]; // [sp+10h] [bp-30h]@7
10   __int64 v10; // [sp+28h] [bp-18h]@1
11
12   v10 = *MK_FP(__FS__, 40LL);
13   LODWORD(v6) = 0;
14   while ( (signed int)v6 < strlen(s) )
15   {
16     if ( v6 & 1 )
17       v3 = 1;
18     else
19       v3 = -1;
20     *(&t + (signed int)v6 + 10) = s[(signed __int64)(signed int)v6] + v3;
21     LODWORD(v6) = v6 + 1;
22   }
23   strcpy(filename, "/tmp/flag.txt");
24   stream = fopen(filename, "w");
25   fprintf(stream, "%s\n", u, v6);
26   for ( i = 0; i < strlen(&t); ++i )
27   {
28     fseek(stream, p[i], 0);
29     fputc(*(&t + p[i]), stream);
30     fseek(stream, 0LL, 0);
31     fprintf(stream, "%s\n", u);
32   }
33   fclose(stream);
34   remove(filename);
35   result = 0;
36   v5 = *MK_FP(__FS__, 40LL) ^ v10;
37   return result;
38 }
```


分析一下逻辑，发现就是t的值就是flag。

写个exp就出来了。

三.exp分享

```
s='c61b68366edeb7bdce3c6820314b7498'
v6=0
t=0x53
v3=1
flag=''
i=0
while v6<len(s):
    if(v6&1):
        v3=1
    else:
        v3=-1
    flag+=chr((ord(s[i])+v3))
    v6=v6+1
    i=i+1
print(flag)
```
```


```


这上面脚本跑出来的，就是？的值，同时别忘记了上面的s。