buptctf(北邮网安杯) 2019 re wp

原创

n00bzx 于 2020-04-01 22:07:37 发布 726 收藏

文章标签： wp 安全

本文链接：https://blog.csdn.net/qq_35623926/article/details/105205948

这次比赛是中学生的ctf比赛,所以只有web和re两部分.当时我才高一…也是第一次打ctf,啥都不会…web记得第一题是php的一堆函数绕过,后面两题不会…上午是算法题6题,下午是ctf.算法我爆零…当时有个巨佬,id是FaFaFa,re ak了…当时下午断网了,啥都没做出来…最近又找到了以前的re题,就做了发上来,题目附在下面…好让大家体会体会…

链接:https://download.csdn.net/download/qq_35623926/12285184

re1:

先看字符串窗口,发现成功和失败字符串.然后查找引用,发现未分析的代码引用了它.找到代码开头,按p建立函数,发现字符串赋值后比对.然后把字符串输出即得到flag.

re2:

ida打开到入口点,f5发现调用两个函数.第一个函数进去是输入flag部分,然后发现调用401000处函数.打开401000,发现是乱码.然后看第二处调用,发现程序创建线程,读取401000代码异或后进行线程注入写回去.把代码dump出来写脚本解密下,ida f5打开如下图:

```
 do
 {
   v3 = a1[v2];
   if ( v3 >= 97 && v3 <= 122 )
   {
     v4 = (v3 - 84) % 26 + 97;
ABEL_13:
     a1[v2] = v4;
     goto LABEL_14;
   }
   if ( v3 >= 65 && v3 <= 90 )
   {
     v4 = (v3 - 52) % 26 + 65;
     goto LABEL_13;
   }
   if ( v3 >= 48 && v3 <= 57 )
   {
     v4 = (v3 - 35) % 10 + 48;
     goto LABEL_13;
   }
ABEL_14:
   ++v2;
 }
 while ( v2 < 28 );
 v5 = &v8;
 do
 {
   if ( *v5 != *v1 )
     goto LABEL_24;
```


是rot13加密…

解密,获得flag…

re3:

这个程序加了壳…peid一查,morphine?见都没见过…od打开,把反调试搞掉后,在第一个区段里下执行断点…找到oep,dump出来,再找代码,发现还有一部分是自解压…在virtualalloc上下断点,解压后,发现反调试还没搞干净,搞干净后再次dump…然后拖到ida里

设置偏移后分析,关于flag的代码很清晰了,如下图:

IDA View-A | Pseudocode-B | Pseudocode-A | Strings window | Hex View-1 | Structures | Enums

```
    v3 = 0;
    do
    {
      *(int *)((char *)&v13 + v3) = *(&v15 + v3) | ((*(&v16 + v3) | ((*(&v17 + v3) | (v18[v3] << 8)) << 8)) << 8);
      v3 += 4;
    }
    while ( v3 < 44 );
    v4 = 0;
    v5 = &unk_1E20D8;
    while ( 1 )
    {
      v6 = 0;
      v7 = 0;
      v8 = 0;
      do
      {
        v6 += *(&v13 + v8) * (unsigned __int8)byte_1E2108[v4 + v8];
        v9 = v14[v8] * (unsigned __int8)byte_1E2109[v4 + v8];
        v8 += 2;
        v7 += v9;
      }
      while ( v8 < 10 );
      v10 = v8 >= 11 ? 0 : *(&v13 + v8) * (unsigned __int8)byte_1E2108[v4 + v8];
      if ( v10 + v7 + v6 != *v5 )
        break;
      ++v5;
      v4 += 11;
      if ( (signed int)v5 >= (signed int)&unk_1E2104 )
        goto LABEL_15;
    }
  }
  sub_1E1190(aSorryTryAgain, v12);
LABEL_15:
  dword_1E205C(0);
  return 0;
}
```

000010B8 sub_1E1000:41 (1E10B8)

使用z3约束器求解,python代码如下:

```
from z3 import *
f=open("./Dumped.bin","rb")
def readdword(arr,index):
        return (arr[index])|(arr[index+1]<<8)|(arr[index+2]<<16)|(arr[index+3]<<24)
def writedword(arr,start,data):
        arr[start]=data&0xff
        arr[start+1]=(data>>8)&0xff
        arr[start+2]=(data>>16)&0xff
        arr[start+3]=(data>>24)&0xff
def readfilebyte(o):
        f.seek(o)
        return int.from_bytes(f.read(1),byteorder="little")
def readfiledword(o):
        f.seek(o)
        a=int.from_bytes(f.read(1),byteorder="little")
        b=int.from_bytes(f.read(1),byteorder="little")
        c=int.from_bytes(f.read(1),byteorder="little")
        d=int.from_bytes(f.read(1),byteorder="little")
        return a|(b<<8)|(c<<16)|(d<<24)
offset=0x1e0000
v5=0x1e20d8
solver = Solver()
flag = [BitVec('flag%d'%i,32) for i in range(44)]
for i in range(44):
        solver.add(flag[i]>=32)
        solver.add(flag[i]<=127)
v4=0
while True:
    v6=0
    v7=0
    v8=0
    while True:
        v6+=readdword(flag,v8*4)*readfilebyte(0x1e2108-offset+v4+v8)
        v9=readdword(flag,v8*4+4)*readfilebyte(0x1e2109-offset+v4+v8)
        v8+=2
        v7+=v9
        if v8>=10:
            break
    v10=readdword(flag,v8*4)*readfilebyte(0x1e2108-offset+v4+v8)
    solver.add(v10+v7+v6==readfiledword(v5-offset))
    v5+=4
    v4+=11
    if v5>=0x1e2104:
        break
if solver.check() == sat:
 m = solver.model()
 s = []
 for i in range(44):
  s.append(m[flag[i]].as_long())
 print(bytes(s))
f.close()
```

运行得到flag:BUPT{Solving_C0n5tra1nts_lSvery_1ntere5t1ng}