

virtual_world

发表于 2021-05-25 更新于 2021-05-26 分类于 [Challenge](#) , [2021](#) , [第四届红帽杯网络安全大赛](#) , [Reverse](#)
[Challenge](#) | [2021](#) | [第四届红帽杯网络安全大赛](#) | [Reverse](#) | [virtual_world](#)

[点击此处](#)获得更好的阅读体验

WriteUp来源

来自[看雪论坛的](#)Delevy

题目描述

题目考点

解题思路

分配指令空间和数据存储空间

□

进入vmp解释器

□

解释器结构

□

由于时间问题没法快速解析指令对应的handler，所以采取IDA指令跟踪的方式
根据对handler指令分析可以发现

```
1 add ?, ?  
2 imul ?, ?  
3 div ?  
4 movzxeax, byte ptr [?]取单字节数据
```

可能为相关运算操作指令

□

□

□

□

使用IDA指令跟踪可以快速分析指令执行，当数据编码完成后开始对比数据

□

0x2B为输入数据编码后的结果

□

直接nop掉跳转指令，运行获取全部对比数据，得到数据

```
1 612a725f7e736dda72df43ab72bf4de96bc965
```

□

```

1 import binascii
2 import z3
3
4 z = z3.Solver()
5 right = binascii.unhexlify("612a725f7e736dda72df43ab72bf4de96bc965")
6 flag = [z3.BitVec("a%d" % _, 32) for _ in range(0x13)]
7 buff = []
8 temp = flag[0]
9
10 for _ in range(len(flag)):
11     try:
12         temp = temp * 3 + flag[_ + 1]
13     except IndexError:
14         temp = temp * 3
15 temp %= (0x1000 + 1)
16 temp &= 0xFF
17 buff.append(flag[0] ^ temp ^ len(flag))
18 temp1 = temp + len(flag)
19 for _ in range(1, len(flag)):
20     buff.append(flag[_] ^ temp1 ^ buff[_ - 1])
21     temp1 += len(flag) - _
22     temp1 &= 0xFF
23
24 for p, i in enumerate(buff):
25     z.add(z3.simplify(i == right[p]))
26
27 if z.check() == z3.sat:
28     for i in flag:
29         print(chr(z.model()[i].as_long()), end="")

```

Flag

```
1 flag{di3869Fop11Nm}
```

- 本文作者: CTFHub
- 本文链接: <https://writeup.ctfhub.com/Challenge/2021/第四届红帽杯网络安全大赛/Reverse/bEwAA9kC57bz4s57FCfk8P.html>
- 版权声明: 本博客所有文章除特别声明外, 均采用 [BY-NC-SA](#) 许可协议。转载请注明出处!

[#Challenge #2021 #Reverse #第四届红帽杯网络安全大赛](#)
[file_store](#)
[primegame](#)