很有趣的题目，学习angr任重道远啊

一开始要去很多花指令，但是花指令也比较好去

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

```
377     *(_BYTE *)(v5 + k + 5) ^= 0x6Au;
378   for ( l = 0LL; l != 32; ++l )
379     *(_BYTE *)(v5 + l + 5) ^= 0x59u;
380   for ( m = 0LL; m != 32; ++m )
381     *(_BYTE *)(v5 + m + 5) ^= 0xAu;
382   for ( n = 0LL; n != 32; ++n )
383     *(_BYTE *)(v5 + n + 5) ^= 0xF3u;
384   for ( ii = 0LL; ii != 32; ++ii )
385     *(_BYTE *)(v5 + ii + 5) ^= 0xCAu;
386   for ( jj = 0LL; jj != 32; ++jj )
387     *(_BYTE *)(v5 + jj + 5) ^= 0x3Eu;
388   for ( kk = 0LL; kk != 32; ++kk )
389     *(_BYTE *)(v5 + kk + 5) ^= 0x6Cu;
390   for ( ll = 0LL; ll != 32; ++ll )
391     *(_BYTE *)(v5 + ll + 5) ^= 0x4Fu;
392   for ( mm = 0LL; mm != 32; ++mm )
393     *(_BYTE *)(v5 + mm + 5) ^= 0x24u;
394   for ( nn = 0LL; nn != 32; ++nn )
395     *(_BYTE *)(v5 + nn + 5) ^= 0x83u;
396   for ( i1 = 0LL; i1 != 32; ++i1 )
397     *(_BYTE *)(v5 + i1 + 5) ^= 0xC4u;
398   for ( i2 = 0LL; i2 != 32; ++i2 )
399     *(_BYTE *)(v5 + i2 + 5) ^= 0x53u;
400   for ( i3 = 0LL; i3 != 32; ++i3 )
401     *(_BYTE *)(v5 + i3 + 5) ^= 4u;
402   for ( i4 = 0LL; i4 != 32; ++i4 )
403     *(_BYTE *)(v5 + i4 + 5) ^= 0x9Eu;
```

很多行这种东西

懂了，angr跑一波

```
from typing import AnyStr
import angr
import claripy

bin_path = 'attachment'
p = angr.Project(bin_path,load_options={"auto_load_libs": False})
flag = claripy.BVS('flag',8*38)

start_addr = 0x400605
init_state = p.factory.blank_state(addr=start_addr)

buffer = init_state.regs.rsp-0x100
flag_addr = buffer+0x50
init_state.regs.rsp = buffer
init_state.memory.store(flag_addr,flag,38)

init_state.regs.rdx = flag_addr
init_state.regs.rdi = flag_addr+5

sm= p.factory.simgr(init_state)
sm.explore(find=0x401db3)

if sm.found:
    answer = sm.found[0]
    print(answer.solver.eval(flag,cast_to=bytes))
```

```
n attachment (0x400616))
WARNING | 2021-08-16 10:21:02,435 | angr.storage.memory_mixins.default_
emory at 0x7fffffffffeff00 with 8 unconstrained bytes referenced from 0
t__+0x10d in attachment (0x4005bd))
WARNING | 2021-08-16 10:21:02,436 | angr.storage.memory_mixins.default_
emory at 0x7fffffffffeff08 with 8 unconstrained bytes referenced from 0
t__+0x10e in attachment (0x4005be))
WARNING | 2021-08-16 10:21:02,637 | angr.engines.successors | Exit stat
solutions. Likely unconstrained; skipping. <BV64 mem_7fffffffffeff08_3_
b'\x00\x00\x00\x00\x001dc20f6e3d497d15cef47d9a66d6f1af\x00'
```