RicknMorty

发表于 2021-01-09 分类于 Challenge， 2020， CSICTF， Reverse
Challenge | 2020 | CSICTF | Reverse | RicknMorty

点击此处获得更好的阅读体验

WriteUp来源

https://dunsp4rce.github.io/csictf-2020/reversing/2020/07/22/RicknMorty.html

by `AnandSaminathan`

题目描述

> *Rick has been captured by the council of ricks and in this dimmention morty has to save him, the chamber holding rick needs a key . Can you help him find the key ?*

题目考点

解题思路

On decompiling using Ghidra:

```
ulong function1(uint param_1,uint param_2) {
  uint local_10;
  uint local_c;

  local_c = 0;
  local_10 = 1;
  while ((local_10 <= param_1 || (local_10 <= param_2))) {
    if ((param_1 % local_10 == 0) && (param_2 % local_10 == 0)) {
      local_c = local_10;
    }
    local_10 = local_10 + 1;
  }
  return (ulong)local_c;
}
long function2(uint param_1) {
  long lVar1;

  if (param_1 == 0) {
    lVar1 = 1;
  }
  else {
    lVar1 = function2(param_1 - 1);
    lVar1 = lVar1 * (ulong)param_1;
  }
  return lVar1;
}
undefined8 main(void) {
  int iVar1;
  time_t tVar2;
  ulong uVar3;
  long lVar4;
  int local_4c;
  time_t local_48;
  time_t local_40;
  time_t local_38;
  uint local_30;
  uint local_2c;
  char *local_28;
  int local_20;
  int local_1c;

  setbuf(stdout,(char *)0x0);
  setbuf(stdin,(char *)0x0);
  setbuf(stderr,(char *)0x0);
  tVar2 = time(&local_38);
  srand((uint)tVar2);
  time(&local_40);
  local_1c = 1;
  local_20 = 0;
  while( true ) {
    iVar1 = rand();
    if (iVar1 % 400 + 100 <= local_20) break;
    iVar1 = rand();
    local_2c = iVar1 % 10 + 6;
    iVar1 = rand();
    local_30 = iVar1 % 10 + 6;
    printf("%d %d",(ulong)local_2c,(ulong)local_30);
    __isoc99_scanf();
    uVar3 = function1(local_2c,local_30);
    lVar4 = function2((int)uVar3 + 3);
    if ((long)local_4c != lVar4) {
      local_1c = 0;
    }
    local_20 = local_20 + 1;
  }
  time(&local_48);
  local_28 = (char *)(double)(local_48 - local_40);
  printf(local_28,"fun() took %f seconds to execute \n");
  if ((local_1c == 1) && ((double)local_28 <= 5.00000000)) {
    system("cat flag.txt");
  }
  return 0;
}
```

In summary, the `main` function has a while loop which on each iteration prints two random numbers (`a` and `b`), then it accepts an input `x` and checks if `function2(function1(a, b) + 3)`. On closer inspection, it is clear that `function1` is `gcd` and `function2` is factorial. So we have to keep reading inputs and provide the right answers to print the flag. This cannot be done manually because there's a time check, so had to use pwntools:

```
from pwn import *
from math import gcd, factorial
io = remote('chall.csivit.com', 30827)
while io.can_recv(1) == True:
    inp = io.recvline()
    inp = inp.decode()
    if inp.split(' ')[0] == 'fun()':
        break
    a, b = inp.split(' ')
    a = int(a)
    b = int(b)
    io.sendline(str(factorial(gcd(a,b) + 3)))
print(io.recvall())
```

Flag

```
csictf{h3_7u2n3d_h1m531f_1n70_4_p1ck13}
```


 # Challenge # 2020 # Reverse # CSICTF
Scrambled-Eggs
Blaise