攻防世界 Crypto高手进阶区 7分题 Decrypt-It-easy

原创

思源湖的鱼 于 2021-02-03 20:57:47 发布 299 收藏

分类专栏： ctf 文章标签： ctf 攻防世界 crypto

本文链接：https://blog.csdn.net/weixin_44604541/article/details/113613885

前言

继续ctf的旅程

攻防世界Crypto高手进阶区的7分题

本篇是Decrypt-It-easy的writeup

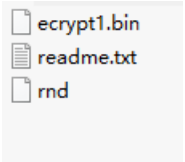
发现攻防世界的题目分数是动态的

就仅以做题时的分数为准了

解题过程

题目描述

得到三个文件



readme里写着`$ ./rnd crypt1.png ecrypt1.bin`

把rnd扔进ida

```
int __cdecl main(int a1, char **a2)
{
  unsigned int v3; // eax
  FILE *v4; // [esp+10h] [ebp-10h]
  FILE *v5; // [esp+14h] [ebp-Ch]
  char ptr; // [esp+1Fh] [ebp-1h]

  if ( a1 <= 2 )
    return 1;
  v3 = time(0);
  srand(v3);
  v4 = fopen(a2[1], "rb");
  v5 = fopen(a2[2], "wb");
  while ( fread(&ptr, 1u, 1u, v4) == 1 )
  {
    ptr ^= rand();
    fwrite(&ptr, 1u, 1u, v5);
  }
  fclose(v4);
  fclose(v5);
  return 0;
}
```


内容很简单
输入一个文件，每个字节异或一个随机数，再输出来

由于输入文件是png
其文件头固定

可以爆破

先取得时间戳

```
kalifisher:~/ctf$ stat ecrypt1.bin
  文件：ecrypt1.bin
  大小：45989           块：96         IO 块：4096   普通文件
设备：801h/2049d        Inode：1349485     硬链接：1
权限：(0600/-rw-------)  Uid：( 1000/       )   Gid：( 1000/
最近访问：2021-02-03 16:59:56.934210000 +0800
最近更改：2014-11-22 22:46:30.000000000 +0800
最近改动：2021-02-03 16:59:56.934210907 +0800
创建时间：-
kalifisher:~/ctf$ stat --printf=%Y  ecrypt1.bin
1416667590  kalifisher:~/ctf$
```

然后爆破

```
#include <stdio.h>
#include <stdlib.h>

int main(int argc, char *argv[]) {
  FILE *cipher = fopen(argv[1], "rb");
  FILE *plain = fopen(argv[2], "wb");
  unsigned int seed = atoi(argv[3]);
  int c;

  srand(seed);
  c = (fgetc(cipher) & 0xff) ^ (rand() & 0xff);
  while (!feof(cipher)) {
    fputc(c, plain);
    c = (fgetc(cipher) & 0xff) ^ (rand() & 0xff);
  }
  fclose(plain);
  fclose(cipher);
}
```

```
kalifisher:~/ctf$ gcc easydecode.c -o easydecode
kalifisher:~/ctf$ ./easydecode ecrypt1.bin crypt1.png 1416667590
```

得到图片

$$N = pq$$
$$C = M(M + B)\ mod\ N$$

$$N = B8AE199365$$
$$B = FFEEE$$
$$C = 8D5051562B$$

$$N = B86E78C811$$
$$B = FFFEE$$
$$C = 5FFA0AC1A2$$

$$N = 7BD4071E55$$
$$B = FEFEF$$
$$C = 6008DDF867$$

解一个类似RSA的加密

陷入困境
去查了查wp
SECCON 2014 quals # Crypto – Decrypt it (Easy)

在已知flag格式：`SECCON{...}`
的情况下做个试验

```
N, B, FLAG = 0xB8AE199365, 0xFFEEE, 'SECCON{'
for i in range(1, len(FLAG)+1):
 M = int(FLAG[0:i].encode('hex'), 16)
 print FLAG[0:i]+'\t'+hex(M * (M + B) % N)
```

```
S       0x52fc213
SE      0x54f0cb0bf
SEC     0x8c0ad9b877
SECC    0x704d68c1fb
SECCO   0x8d5051562bL
SECCON  0x2339eed575L
SECCON{ 0x1bce931b16L
```

发现最多5位字符

那就有

```
N, B = 0xB86E78C811, 0xFFFEE
for i in range(32, 127):
  for j in range(32, 127):
    for k in range(32, 127):
      M = int(('N{' + chr(i) + chr(j) + chr(k)).encode('hex'), 16)
      if ((M * (M + B) % N) == 0x5FFA0AC1A2):
        print 'N{' + chr(i) + chr(j) + chr(k)
N, B = 0x7BD4071E55, 0xFEFEF
for i in range(32, 127):
  for j in range(32, 127):
    for k in range(32, 127):
      for l in range(32, 127):
        M = int((chr(i) + chr(j) + chr(k) + chr(l) + '}').encode('hex'), 16)
        if ((M * (M + B) % N) == 0x6008DDF867):
          print chr(i) + chr(j) + chr(k) + chr(l) + '}'
```

得到flag：`SECCON{Ra_b1_N}`

大神还给了个优美脚本

```
def mul_inv(a, b):
  b0 = b
  x0, x1 = 0, 1
  if b == 1: return 1
  while a > 1:
    q = a / b
    a, b = b, a%b
    x0, x1 = x1 - q * x0, x0
  if x1 < 0: x1 += b0
  return x1

def chinese_remainder(n, a, lena):
  p = i = prod = 1; sm = 0
  for i in range(lena): prod *= n[i]
  for i in range(lena):
    p = prod / n[i]
    sm += a[i] * mul_inv(p, n[i]) * p
  return sm % prod

def bruteforce(p, q, C, B):
  for Mp in [m for m in xrange(p) if (C % p) == (m * (m + B) % p)]:
    for Mq in [m for m in xrange(q) if (C % q) == (m * (m + B) % q)]:
      print ('%x' % chinese_remainder([p, q], [Mp, Mq], 2)).decode('hex')
```

结语

学到了