HGAME WEEK4 WriteUp

hgame 2019终于结束了！第一次参加ctf，收获了超级多！感谢学长们（づ￣3￣）づ╭❤~

RE

real

happyvm

这两道题因为我是一边做一边写的笔记，所以很长。这里把链接放一下

www.danisjiang.xyz/1

如果在23号8点前看的话是有密码的，密码是112233

MISC

暗藏玄机

一开始得到两张外表一样的文件

看了一下大小，一个是900多kb，一个是300多kb，说明这两张图片中有猫腻

一开始以为可以用compare命令，但是得到的这个图片有点不太理解意图

所以又去搜了搜，发现有一个盲水印的东西，与这个题目的条件很符合，就是图片长得一样，但是大小不同

github上有现成的脚本

最后得到盲水印

得到flag

Warmup

```
danis@ubuntu:~/MISC$ file 1.gif
1.gif: Mini DuMP crash report, 12 streams, Mon Feb 11 14:20:58 2019, 0x1826 type
danis@ubuntu:~/MISC$
```

得到的gif文件其实是一个Mini Dump文件，一开始以为这个内存取证类题目时用volatility做的，然而

```
danis@ubuntu:~/MISC$ volatility imageinfo -f 1.dmp
Volatility Foundation Volatility Framework 2.6
INFO    : volatility.debug    : Determining profile based on KDBG search...
          Suggested Profile(s) : No suggestion (Instantiated with no profile)
                     AS Layer1 : FileAddressSpace (/home/danis/MISC/1.dmp)
                      PAE type : No PAE
```

并不能，后来添加了hint，得知Mini Dump文件其实是通过mimikatz生成的？反正可以用mimikatz来查看

```
Authentication Id : 0 ; 2353730 (00000000:0023ea42)
Session           : Interactive from 2
User Name         : Hgame
Domain            : xyf-PC
Logon Server      : XYF-PC
Logon Time        : 2019/2/11 22:02:44
SID               : S-1-5-21-373264735-3061158248-1611926753-1003
        msv :
         [00000003] Primary
         * Username : Hgame
         * Domain   : xyf-PC
         * LM       : 758ff83c96bcac17aad3b435b51404ee
         * NTLM     : e527b386483119c5218d9bb836109739
         * SHA1     : ca17a8c02628f662f88499e48d1b3e9398bef1ff
        tspkg :
         * Username : Hgame
         * Domain   : xyf-PC
         * Password : LOSER
        wdigest :
         * Username : Hgame
         * Domain   : xyf-PC
         * Password : LOSER
        kerberos :
         * Username : Hgame
         * Domain   : xyf-PC
         * Password : LOSER
        ssp :
        credman :
```

password就是LOSER，flag是password的sha256

LOSER

在线加密 在线解密

sha256（LOSER） = dd6dffcd56b77597157ac6c1beb514aa4c59d033098f806d88df89245824d3f5

得到flag

CRYPTO

easy_rsa

共模攻击，但是e1和e2不互质，下面是我对这道题的理解

$$c1 = (m^3)^{e1} \bmod n$$

$$c2 = (m^3)^{e2} \bmod n$$

$$(c1^{-1})^{s1} * (c2)^{s2} = M^3 \bmod n$$

附上脚本

```
import libnum
import gmpy2

def main():
  n =
0x9439682bf1b4ab48c43c524778c579cc844b60872275725c1dc893b5bcb358b9f136e4dab2a06318bb0c80e20
2a14bc54ea334519bec023934e01e9378abf329893f3870979e9f2f2be8fff4df931216a77007a2509f49f697bf
286285e97fac5dc6e4a164b5c2cc430887b18136437ba67777bda05aafdeaf918221c812b4c7d1665238f84ab0f
ab7a77fcae92a0596e58343be7a8e6e75a5017c63a67eb11964970659cd6110e9ec6502288e9e443d86229ef236
4dfecb63e2d90993a75356854eb874797340eece1b19974e86bee07019610467d44ec595e04af02b574a97fa98b
db2e779871c804219cab715f4a80fef7f8fb52251d86077560b39c1c2a1
  c1 =
0x7c7f315a3ebbe305c1ad8bd2f73b1bb8e300912b6b8ba1b331ac2419d3da5a9a605fd62915c11f8921c450525
d2efda7d48f1e503041498f4f0676760b43c770ff2968bd942c7ef95e401dd7facbd4e5404a0ed3ad96ae505f87
c4e12439a2da636f047d84b1256c0e363f63373732cbaf24bda22d931d001dcca124f5a19f9e28608ebd90161e7
28b782eb67deeba4cc81b6df4e7ee29a156f51a0e5148618c6e81c31a91036c982debd1897e6f3c1e5e248789c9
33a4bf30d0721a18ab8708d827858b77c1a020764550a7fe2ebd48b6848d9c4d211fd853b7a02a859fa0c721606
75d832c94e0e43355363a2166b3d41b8137100c18841e34ff52786867d
  c2 =
0xf3a8b9b739196ba270c8896bd3806e9907fca2592d28385ef24afadc2a408b7942214dad5b9e14808ab988fb1
5fbd93e725edcc0509ab0dd1656557019ae93c38031d2a7c84895ee3da1150eda04cd2815ee3debaa7c2651b626
39f785f6cabf83f93bf3cce7778ab369631ea6145438c3cd4d93d6f2759be3cc187651a33b3cc4c3b4776044771
43c32dfff62461fdfd9f8aa879257489bbf977417ce0fbe89e3f2464475624aafef57dd9ea60339793c69b53ca7
1d745d626f45e6a7beb9fcbd9d1a259433d36139345b7bb4f392e78f1b5be0d2c56ad50767ee851fac670946356
b3c05d0605bf243b89c7e683cc75030b71633632fb95c84075201352d6
```

```
  e1 = 0x33240/3
  e2 = 0x3e4f/3
  m = common_modulus(n,e1,e2,c1,c2)
  print gmpy2.iroot(m,3)

def common_modulus(n, e1, e2, c1, c2):

    assert (libnum.gcd(e1, e2) == 1)
    _, s1, s2 = gmpy2.gcdext(e1, e2)

    m = pow(c1, s1, n) if s1 > 0 else pow(gmpy2.invert(c1, n), -s1, n)
    m *= pow(c2, s2, n) if s2 > 0 else pow(gmpy2.invert(c2, n), -s2, n)
    return m % n

main()
```

```
danis@ubuntu:~/CRYPTO$ python ./rsa2.py
(mpz(59594981651654789), True)
danis@ubuntu:~/CRYPTO$
```

得到flag