

**NAMA TIM : [Cyber Security Trunojoyo]**

|  |  |
| --- | --- |
| **Ketua Tim** | |
| **1.** | Wijanarko Putra Rajeb |
| **Member** | |
| **1.** | Muhamad Hendrik Wicaksono |
| **2.** | Muhammad Zaelani |

Universitas Trunojoyo Madura

KKSI 2019 UMUM - Surabaya

**Daftar Isi**

**Pwn**

Pwn3 [ 50 Point ]

**Misc**

Misc1 [ 50 Point ]

Misc2 [ 150 Point ]

Misc3 [ 200 Point ]

**Reversing**

Rev1 [ 50 Point ]

**Crypto**

Crypt1 [ 200 Point ]

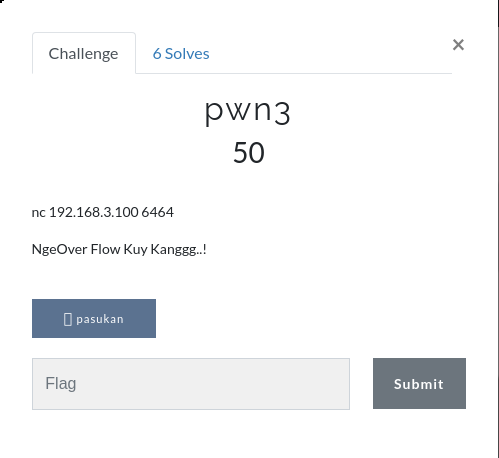
**Web**

Web1 [ 100 Point ]

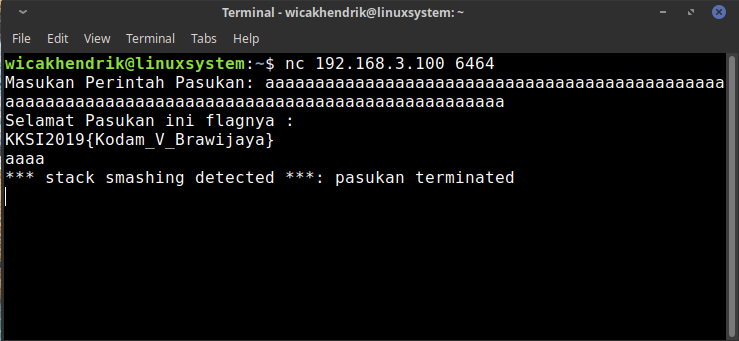
**Pwn**

Pwn3 [ 50 Point ]

Disediakan soal berikut:



Diberikan sebuah soal diatas dengan hint buffer. Maka kami mencoba dengan menginputkan banyak huruf maka didapatkan.



**FLAG : KKSI2019{Kodam\_V\_Brawijaya}**

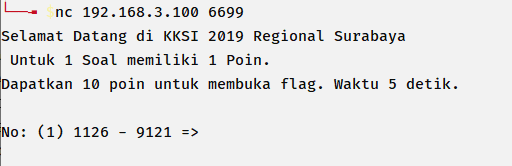
**Misc**

Misc1 [ 50 Point ]

Disediakan soal berikut.



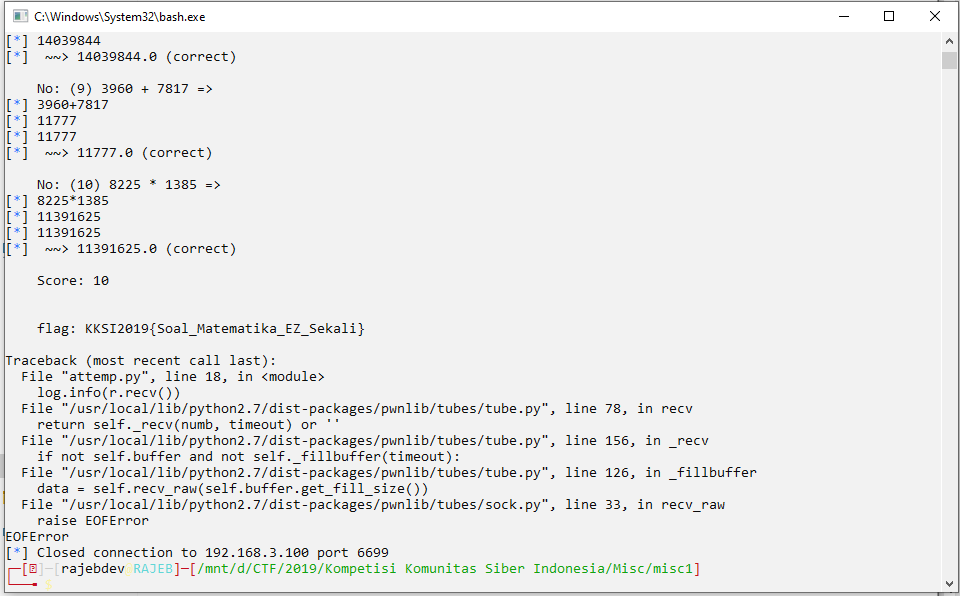
Disediakan service dengan tampilan seperti berikut :



Dalam 5 detik kita diminta untuk menjawab soal. Namun hal itu tidak muungkin jika dilakukan manual, maka dilakukan otomasi dengan menggunakan script python berikut.

|  |
| --- |
| from pwn import \*  r = remote("192.168.3.100", 6699)  while True:  try:  angka = r.recvuntil("=> ")  log.info(angka)  a1 = angka.split()[-4]  op = angka.split()[-3]  a2 = angka.split()[-2]  log.info(a1+op+a2)  ans = str(eval(a1+op+a2))  log.info(ans)  r.sendline(ans)  log.info(ans)  except:  log.info(r.recv()) |

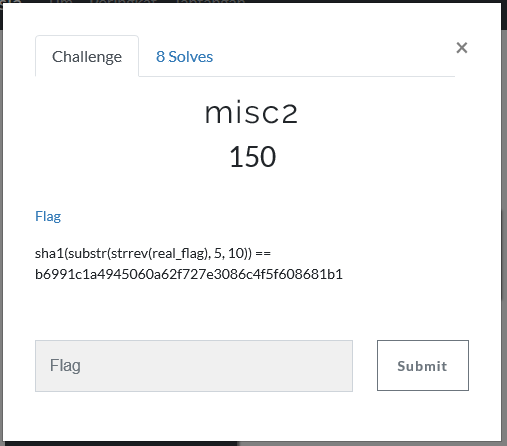
Maka didapatkan Flag seperti berikut:



**FLAG : KKSI2019{Soal\_Matematika\_EZ\_Sekali}**

Misc2 [ 150 Point ]

Disediakan soal berikut.



Disediakan sebuah file flag\_surabaya.txt dan soal seperti diatas. Disini kita diminta untuk melakukan brute forcing pada text dengan beberapa algoritma. Berikut adalah script solvernya.

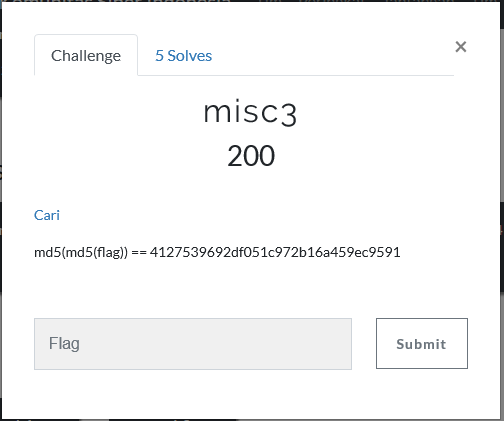
|  |
| --- |
| flag = open("flag\_surabaya.txt", "r").read().split('\n')  import base64  import hashlib  target = 'b6991c1a4945060a62f727e3086c4f5f608681b1'  for find in flag[:-1]:  find = find[::-1]  find = base64.b64decode(find)  flags = find  find = find[::-1]  find = find[5:15]  if hashlib.sha1(find).hexdigest() == target:  print(flags) |

Ketika dijalankan script mengasilkan flag.

**FLAG : KKSI2019{gmvW6EVRUd}**

Misc3 [ 200 Point ]

Disediakan soal berikut.



Disediakan file find\_fix.txt yang berisi kumpulan md5 dengan 2 bagian awal dan akhir kosong. Artinya kita harus mencoba satu persatu dengan menggunakan list angka dan huruf hexa. Berikut adalah script untuk melakukan brute forcing. Soal ini mirip dengan soal penyisihan kemaren juga. Jadi agak mudah untuk dilakukan solve.

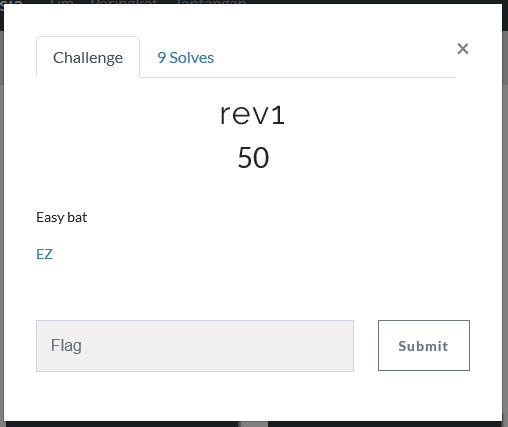
|  |
| --- |
| import hashlib  flag = open("find\_fix.txt", "r").read().replace('?', '{}').split('\n')  liss = '1234567890abcdef'  target = '4127539692df051c972b16a459ec9591'  for i in liss:  for j in liss:  for find in flag:  payload = find.format(i,j)  if hashlib.md5(hashlib.md5(payload).hexdigest()).hexdigest() == target:  print "FLAG : KKSI2019{%s}" % payload |

**FLAG : KKSI2019{26c134c8e04c1e54c1d0893b3d7de4b0}**

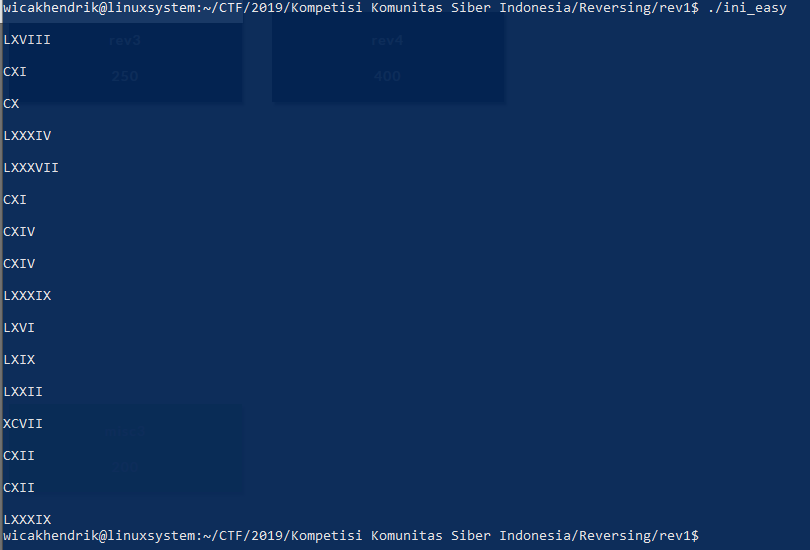
**Reversing**

Rev1 [ 50 Point ]

Disedikan Soal Berikut:



Disediakan sebuah binary yang ketika dilankan muncul



Kemudian dilakukan decode pada romawi dengan script berikut

|  |
| --- |
| def transform\_roman\_numeral\_to\_number(roman\_numeral):  roman\_char\_dict = {'I': 1, 'V': 5, 'X': 10, 'L': 50, 'C': 100, 'D': 500, 'M': 1000}  res = 0  for i in range(0, len(roman\_numeral)):  if i == 0 or roman\_char\_dict[roman\_numeral[i]] <= roman\_char\_dict[roman\_numeral[i - 1]]:  res += roman\_char\_dict[roman\_numeral[i]]  else:  res += roman\_char\_dict[roman\_numeral[i]] - 2 \* roman\_char\_dict[roman\_numeral[i - 1]]  return res  text = 'LXVIII CXI CX LXXXIV LXXXVII CXI CXIV CXIV LXXXIX LXVI LXIX LXXII XCVII CXII CXII LXXXIX'  flag = ''  for i in text.split():  flag += chr(transform\_roman\_numeral\_to\_number(i))  print("KKSI2019{%s}"%flag) |

Maka didapatkan flag

FLAG : KKSI2019{DonTWorrYBEHappY}

Cryto 1

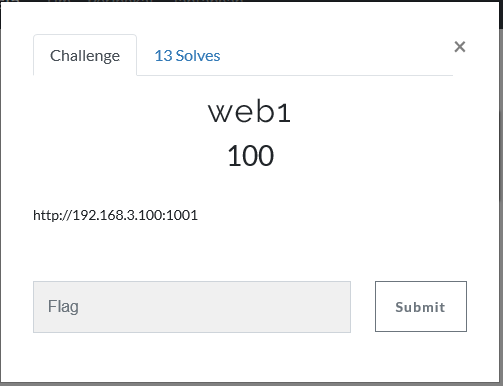


Disediakan file python enkrisi. Kemudian kami Analisa dan kami lakukan decrypt dengan script berikut :

|  |
| --- |
| enc = [  256,  9223372036854775845,  178,  47,  196,  52,  232,  9223372036854775865,  208,  53,  226,  9223372036854775857,  252,  9223372036854775865,  230,  9223372036854775870,  220,  9223372036854775865,  240,  9223372036854775857,  264,  9223372036854775865,  264,  9223372036854775868,  170,  9223372036854775883,  136,  9223372036854775830,  122,  23,  168,  9223372036854775847,  172,  9223372036854775843,  ][::-1]  def rotl(num, bits=64):  bit = num & (1 << (bits - 1))  num <<= 1  if bit:  num |= 1  num &= 2 \*\* bits - 1  return num  def rotr(num, bits=64):  num &= 2 \*\* bits - 1  bit = num & 1  num >>= 1  if bit:  num |= 1 << (bits - 1)  return num  for i in range(128):  encxor = []  for j in enc:  encxor.append(i ^ j)  # print(encxor)  flagr = []  for x, y in enumerate(encxor):  if x % 2 == 0:  flagr.append(rotl(y))  else:  flagr.append(rotr(y))  # print(flagr)  for y in range(128):  flag = ""  for z in flagr:  try:  flag += chr(z - y)  except:  pass  if "KKSI" in flag:  print(flag) |

**FLAG : KKSI2019{Hey\_you\_can\_solve\_it\_BTW}**

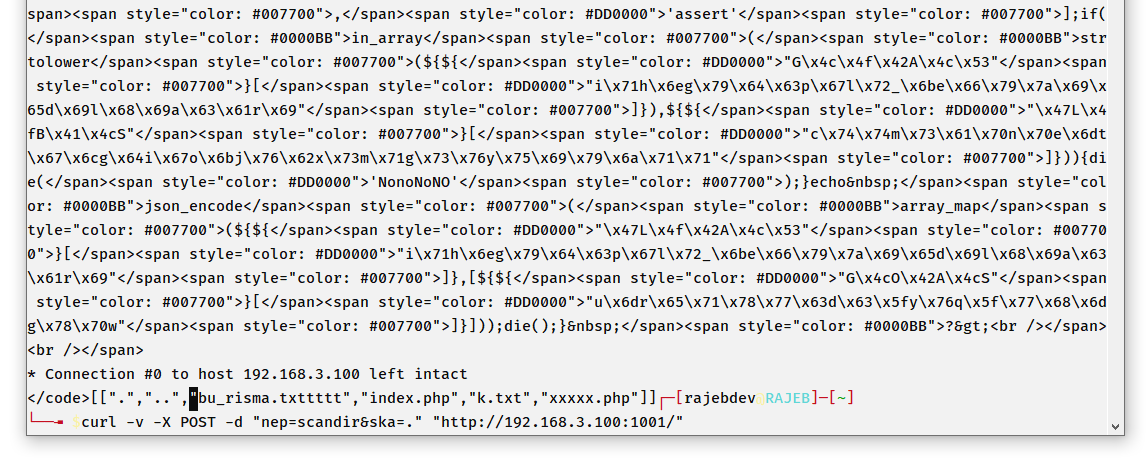
**WEB 1**



Dilakukan post dengan curl

**curl -v -X POST -d "nep=scandir&ska=."** [**http://192.168.3.100:1001/**](http://192.168.3.100:1001/)

Maka didapatkan sebuah list file



Kemudian dilakukan akses <http://192.168.3.100:1001/bu_risma.txttttt>

FLAG : KKSI2019{Aku\_Cinta\_Bu\_Risma}