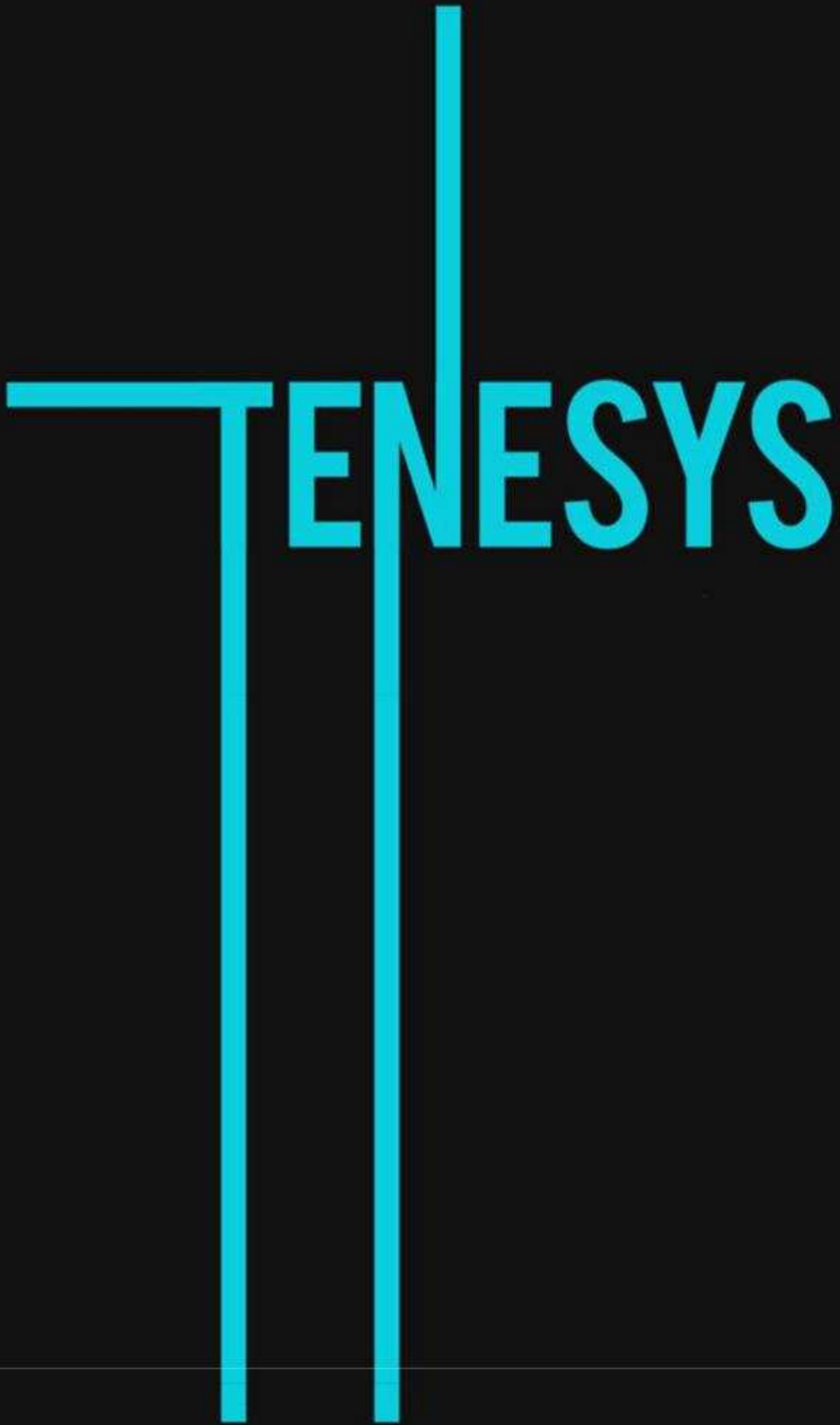

TENESYS


IAM9ROOT
AKNARI
TEKNOKRAT AND SYSTEM SECURITY TENESYS 2019

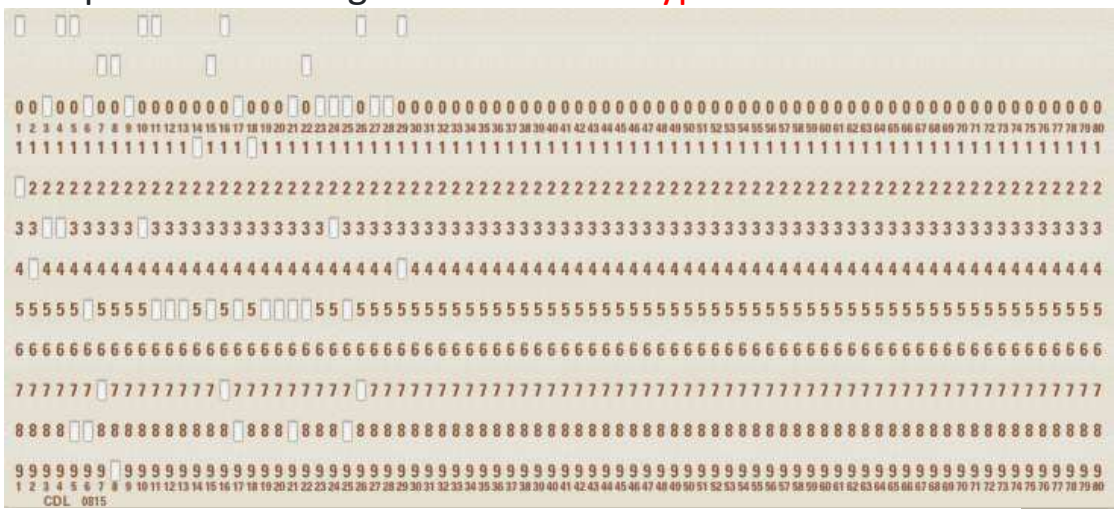
1. Sanity Check – 1 point – Misc –

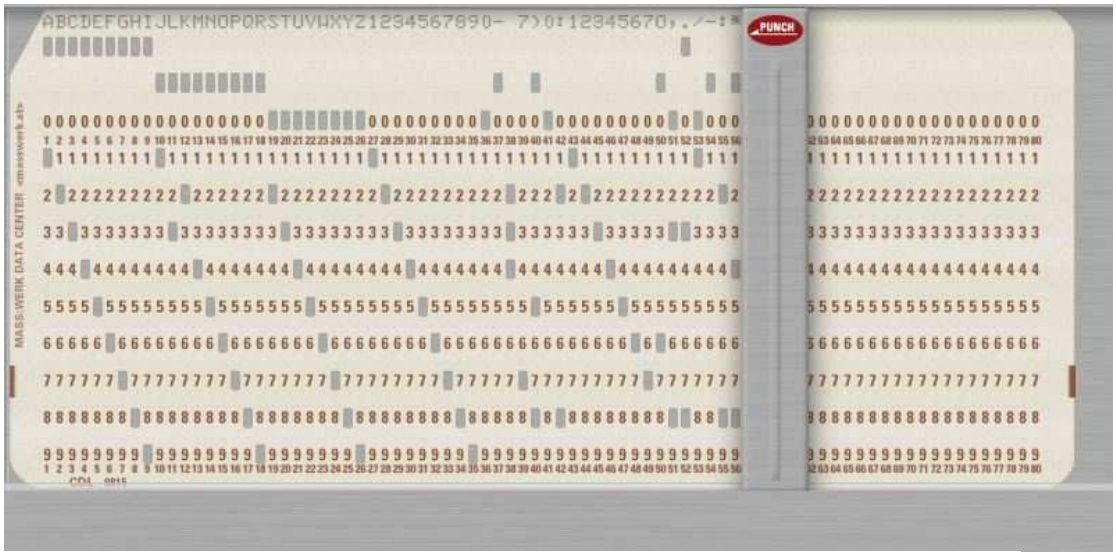
Nothing

- Didapatkan tidak ada apa apa
- Lalu lihat rules
  - Attacking Infrastructure is prohibited, doing so will result in a ban
  - Join our Discord server for the latest information.
  - Points awarded are at our discretion.
  - Reach out to the admins on Discord in case of any problems.
  - Do not share flags.
  - CTF will start on Tue, 02 April 2019, 10:30 IST
  - Thanks for reading rules here's your free flag: encryptCTF{L3t5_H4CK}
- Dan didapatkan flag : **encryptCTF{L3t5_H4CK}**

2. Way Back – 50 point – Misc –

put the message in encryptCTF{}

- Didapatkan sebuah gambar Virtual Keypunch
- 
- Lalu di decode dengan https://www.masswerk.at/keypunch/
- Disini saya melakukan manual dengan membuat dari A sampai dengan selesai



- Lalu ku sesuaikan dengan gambar sebelumnya
- Dan didapatkan flag : **encryptCTF{B4TCH_PR0CE551NG_155_N0T_G00D}**

3. yhpargonagets – 20 point – Steganography–

Find it fi y0u can :p Author: inc0gnito

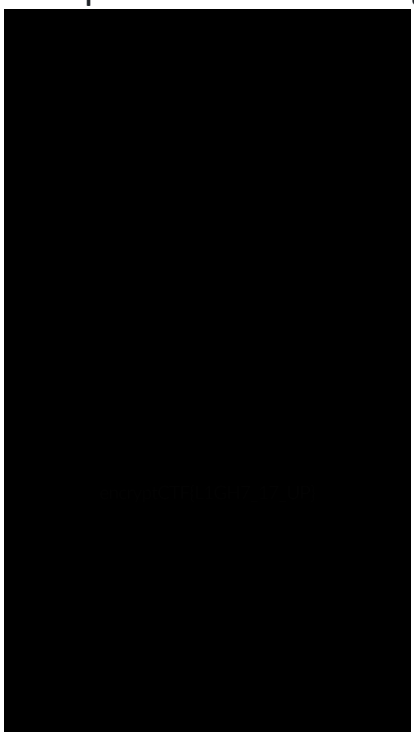
- Didapatkan sebuah gambar

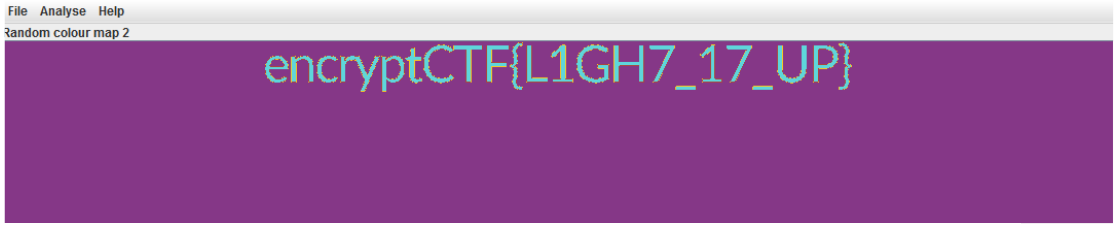
- Lalu pip install steganography

- https://pypi.org/project/steganography/
- Lalu panggil steganography file
- Dan didapatkan flag :
  encryptCTF{pip_in5t411_5teg4n0graphy}

4. Into The Black – 50 point – Steganography –

"My My, Hey Hey....., Rock & Roll is here to stay.. It's better to burn up, Then to fade away...., My My, Hey Hey....." meme Author:@mostwanted002

- Didapatkan sebuah file gambar black
- 

- Lalu disini saya menggunakan stegsolve untuk menggeser warnanya
- 

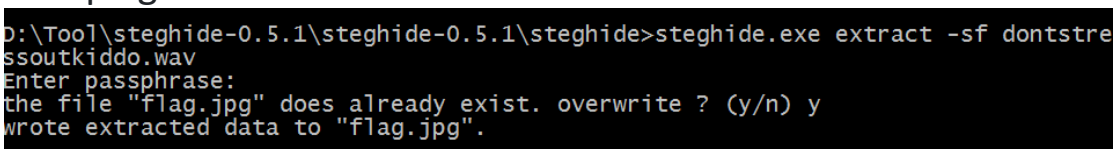
- Dan didapatkan flag : **encryptCTF{L1GH7_17_UP}**

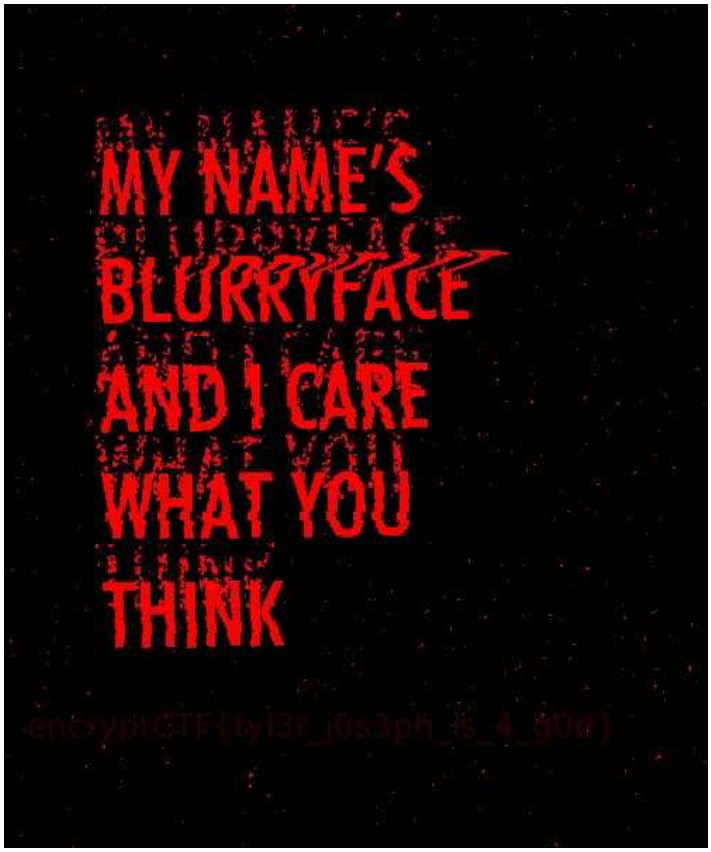
5. Stressed out? – 100 point – Steganography –

Elliot all stressed out from this hack, that hack, saving the world (yeeeeep, sounds about right) was losing his mind when Mr. Robot handed him this song to relax to.

Elliot: It's good. So good, it scratched that part of my mind. The part that doesn't allow good to exist without a condition.

Author: maskofmydisguise

- Didapatkan sebuah file wav
- Lalu disini saya mencoba dengan menggunakan steghide apakah bisa ? ternyata bisa
- Lalu didapatkan password di Tittle yaitu 1_4M_Str3ss3d_0ut
- Lalu masukkan passwordnya dan didapatkan sebuah flag berupa gambar
- 



- Lalu dicerahkan kontrasnya maka akan kelihatan atau bisa distegsolve
- Dan didapatkan flag : **encryptCTF{tyl3r_j0s3ph_is_4_g0d}**

6. Hard Looks – 75 point – Crypto –

Does this look hard?

CipherText: --_--___--_-_-__--_--__--__-_-__--_--___--__--__--_--
-__--__-___--_---__---__-__--_---__--_______--_---__--_-____--_-
____--__--__--_-_-__--_-____--_-____--_--___--_---_--___-___--
_-_-__--_---__--__--__--_-____--__--__--_-_-__--_-_-_--__--___--
__--__--___-__--__--__--_---__--_-_-_--__--___--_-____---_____-
-__--__--_-_____--_-_-__--__-___--_-____--_-____--_-____--_-_--__
___--__--__--__--__--_-___--__-_-__--__--__-_______--_-_-__--
_-_-__--_-____--_---__--_-____---_____--__--_--__--___--__-
___--___-__--_---_--__-__

Author: @mostwanted002

- Didapatkan sebuah ciphertext
- Lalu ubah – menjadi 1 dan _ menjadi 0
- Lalu ternyata biner ada yang kurang

```
11011000 11010100 11011001 10010100 11011000
11001100 11011100 11001000 11011100 11100100
11011100 11000000 11011100 11010000 11010000
11001100 11010100 11010000 11010000 11011000
11011101 10001000 11010100 11011100 11001100
11010000 11001100 11010100 11010101 10011000
11001100 11000100 11001100 11011100 11010101
10011000 11010000 11100000 11001100 11010000
11010100 11001000 11010000 11010000 11010101
10011000 11001100 11001100 11010001 10010100
11001100 11000000 11010100 11010100 11010000
11011100 11010000 11100000 11001101 10011000
11001000 11000100 11011101 100100
```

- Lalu ditambahkan 00 di depan dan didapatkan Hexadecimal

```
00110110 00110101 00110110 01100101 00110110
00110011 00110111 00110010 00110111 00111001
00110111 00110000 00110111 00110100 00110100
00110011 00110101 00110100 00110100 00110110
00110111 01100010 00110101 00110111 00110011
00110100 00110011 00110101 00110101 01100110
00110011 00110001 00110011 00110111 00110101
01100110 00110100 00111000 00110011 00110100
00110101 00110010 00110100 00110100 00110101
01100110 00110011 00110011 00110100 01100101
00110011 00110000 00110101 00110101 00110100
00110111 00110100 00111000 00110011 01100110
00110010 00110001 00110111 01100100
```

```
656e63727970744354467b5734355f31375f483452445f3
34e305547483f217d
```

- Lalu di decode hex to text

```
>>> "656e63727970744354467b5734355f31375f483452445f334e305547483f217d".decode(
'Hex')
'encryptCTF{W45_17_H4RD_3N0UGH?!}'
```

- Dan didapatkan flag : **encryptCTF{W45_17_H4RD_3N0UGH?!}**

7. Rsa_Baby – 100 point – Crypto –

RSA is one of the first public-key cryptosystems and is widely

used for secure data transmission. In such a cryptosystem, the encryption key is public and it is different from the decryption key which is kept secret.

Google up, understand it, and the flag was encrypted with this attached Python Script.

- Didapatkan sebuah

Ciphertext =
1899b6cd310966281b1593a420205588f12ab93af850ad7d9
d810a502f6fe4ad93a58b5bbb747803ba33ac94cc5f227761e
72bdd9857b7b0227f510683596791526b9295b20be39567fc
9a556663e3b0e3fcc5b233e78e38a06b29314d897258fbe15b
037d8ff25d272822571dd98dfa4ee5d066d707149a313ad0c9
3e79b4ee

n=
12896639584745682324232796836643715162628700560457154353002080765348185463443246356750557925507540084680268692376346549839322168386755082407117695374739088192612345473835987918645568185135641426115528380241487388557417214484044788208796961578148633184979831591286939071086573815797450117166560101172338543552 3

p = getPrime(512)

q =
98969843951515664924487488621392623453872977851446373324999664265713980402950871255587801215048348472898280373716439271994046152186233143268514731297699891

e = 65537

- Lalu terdapat p yang belum diketahui
- Karena n = p * q maka untuk mendapatkan p maka n/q
- Lalu running script berikut ini :

```
import gmpy2

def num_to_str(num):
    res = ""
    while num > 0:
        res = chr(num % 256) + res
        num = num / 256
    return res

c =
0x1899b6cd31096281b1593a420205588f12ab93
af850ad7d9d810a502f6fe4ad93a58b5bbb747803
ba33ac94cc5f227761e72bdd9857b7b0227f51068
3596791526b9295b20be39567fc9a556663e3b0e3
fcc5b233e78e38a06b29314d897258fbe15b037d8
ff25d272822571dd98dfa4ee5d066d707149a313a
d0c93e79b4ee
n =
12896639584745682324232796836643715162628
70056045715435300208076534818546344324635
67505579255075400846802686923763465498393
22168386755082407117695374739088192612345
47383598791864556818513564142611552838024
14873885574172144840447882087969615781486
33184979831591286939071086573815797450117
166560101172338543523
q =
98969843951515664924487488621392623453872
97785144637332499664265713980402950871255
55878012150483484728982803737164392719940
46152186233143268514731296998891
p = n/q
e = 65537
t = (p-1)*(q-1)
```

```
d = gmpy2.invert(e,t)
m = pow(c,d,n)

print "Flag: ", num_to_str(m)
```

- Dan didapatkan flag : **encryptCTF{74K1NG_B4BY_S73PS}**
- Makasih bang **fredrica** :D

8. Julius,Q2Flc2FyCg== – 100 point – Crypto –

World of Cryptography is like that Unsolved Rubik's Cube, given to a child that has no idea about it. A new combination at every turn.

Can you solve this one, with weird name?

ciphertext: fYZ7ipGIjFtsXpNLbHdPbXdaam1PS1c5lQ

- Didapatkan sebuah ciphertext
- Lalu saya mencoba decode base64 dengan python ternyata muncul error incorrect padding

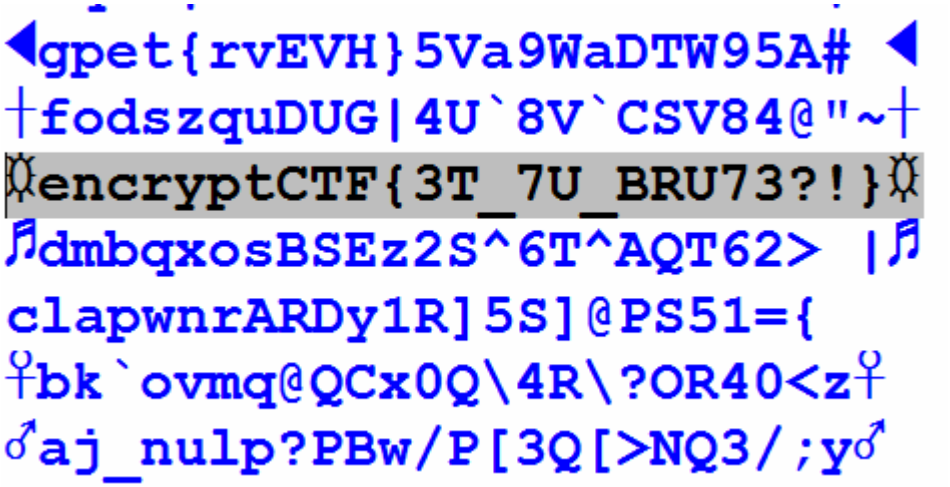
```
>>> "fYZ7ipGIjFtsXpNLbHdPbXdaam1PS1c5lQ".decode('base64')

Traceback (most recent call last):
  File "<pyshell#3>", line 1, in <module>
    "fYZ7ipGIjFtsXpNLbHdPbXdaam1PS1c5lQ".decode('base64')
  File "C:\Python27\lib\encodings\base64_codec.py", line 42, in base64_decode
    output = base64.decodestring(input)
  File "C:\Python27\lib\base64.py", line 321, in decodestring
    return binascii.a2b_base64(s)
Error: Incorrect padding
```

- Lalu saya tambahkan padding dan decode kembali

```
>>> "fYZ7ipGIjFtsXpNLbHdPbXdaam1PS1c5lQ==".decode('base64')
'}\x86{\x8a\x91\x88\x8c[l^\x93KlwOmwZjmOKW9\x95'
```

- Didapatkan ‘}\x86{\x8a\x91\x88\x8c[l^\x93KlwOmwZjmOKW9\x95’
- Lalu saya Brute Force Caesar

```
def caesar_decode():
    decoded =
"'}\x86{\x8a\x91\x88\x8c[l^\x93KlwOmwZjmO
KW9\x95'"
    for k in range(200):
        aw=""
        for i in decoded:
            aw+=chr((ord(i)-k)%256)
        print(aw)
```



- Dan didapatkan flag : **encryptCTF{3T_7U_BRU73?!}**

9. (TopNOTCH)SA – 150 point – Crypto –

This admin's Obsession with RSA is beyond crazy, it's like he's being guided by some people more supreme, the top Notch of 7 Billion....

Anyways, here's the archive, you know the deal. GodSpeed!

Author:@mostwanted002

- Didapatkan sebuah ciphertext, pubkey dan e

Ciphertext :
369ad6199548d8181c26d112d1061008c056f08c753393664
35046a9a8fbf295

-----BEGIN PUBLIC KEY-----
MDswDQYJKoZIhvcNAQEBBQADKgAwJwIgf/0rGqcnR/agG5+
Wd3h7oXKQkz46RmQM
7lU4NDIJq9ECAwEAAQ==
-----END PUBLIC KEY-----

e = 65537

- Lalu disini saya menganalisis pubkey bisa menggunakan openssl atau http://merricx.github.io/enigmator/cryptanalysis/rsa_key_analysis.html
- Lalu didapatkan n

n =
57891041571118599917733172578294383243762455810797917992757930072844611988433

- Lalu disini tinggal mencari p dan q
- Untuk mencari p dan q maka bisa menggunakan http://factordb.com/ masukkan nilai n nya.

p = 194038568404418855662295887732506969011

q = 298348117320990514224871985940356407403

- Lalu running script berikut ini :

```
import gmpy2
```

```
def num_to_str(num):
    res = ""
    while num > 0:
        res = chr(num % 256) + res
        num = num / 256
    return res

c =
0x369ad6199548d8181c26d112d1061008c056f08
c75339366435046a9a8fbf295
n =
57891041571118599917733172578294383243762
45581079791799275793007284461198843 3
p =
19403856840441885566229588773250696901 1
q =
29834811732099051422487198594035640740 3
e = 65537
t = (p-1)*(q-1)
d = gmpy2.invert(e,t)
m = pow(c,d,n)

print "Flag: ", num_to_str(m)
```

- Dan didapatkan flag : **encryptCTF{1%_0F_1%}**

10. AEeeeeS – 200 point **– Crypto –**

... he encrypted the flag Using AES ECB.

the key he gave, is below.

Is he mad?

```ciphertext:
c68145ccbc1bd6228da45a574ad9e29a77ca32376bc1f2a1e4cd
```

66c640450d77```

Author: @mostwanted002

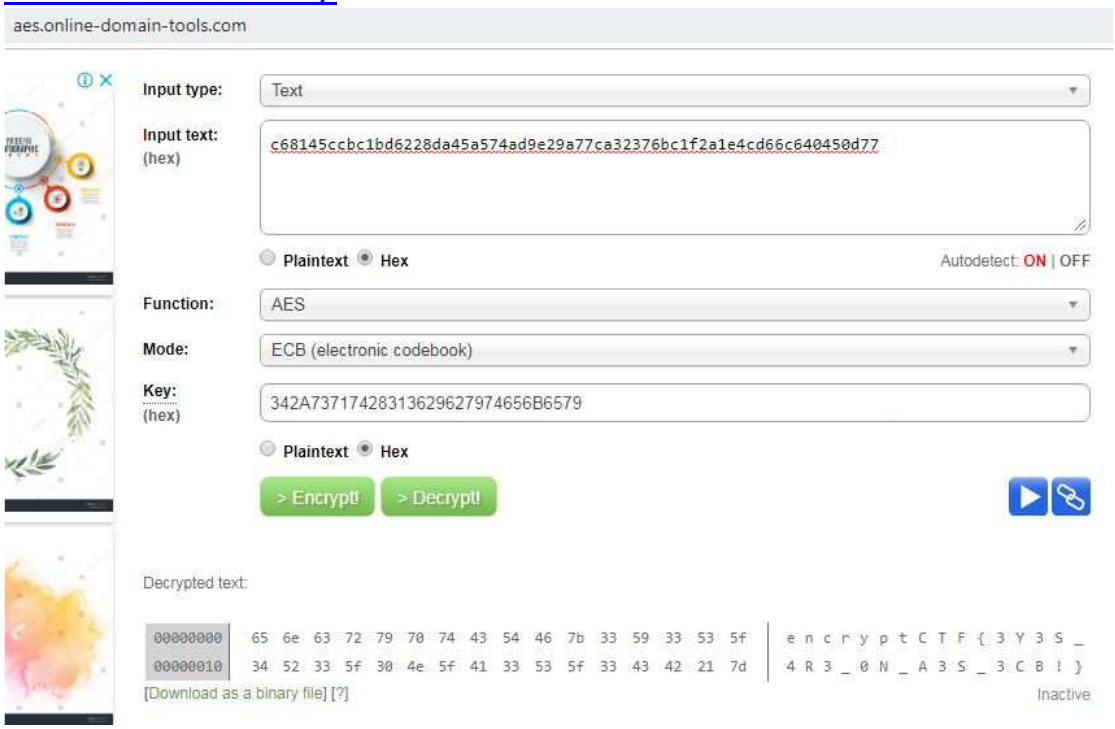
- Didapatkan sebuah ciphertext dan key dan clue AES ECB

Key =
1101000010101001110011011100010111010000101000001
1000100110110001010010110001001111001011101000110
0101011010110110010101111001

- Lalu saya ubah kunci dari biner menjadi hex
- Dan didapatkan hex nya

342A7371742831362962797465 6B6579

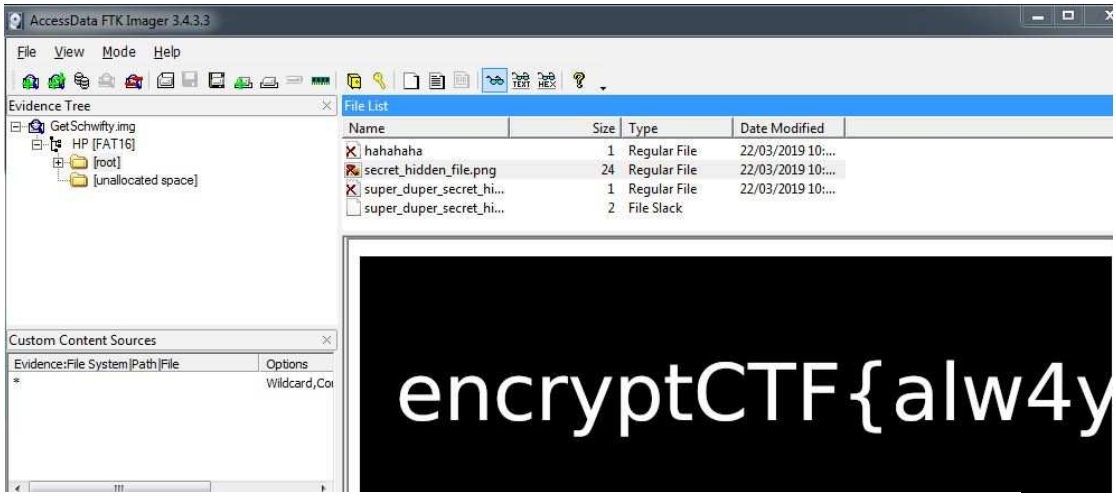
- Lalu saya decrypt menggunakan http://aes.online-domain-tools.com/



- Dan didapatkan flag : **encryptCTF{3Y3S_4R3_0N_A3S_3CB!}**

11. Get Schwifty – 10 point **– Forensics –**

Evil Morty, the first democratically-elected President of the Citadel of Ricks, has killed off twenty-seven known Ricks from various dimensions, as well as capturing, torturing, and enslaving hundreds of Mortys. As a fellow Rick-less Morty, Investigator Rick gives you a file revealing Evil Morty's past and true nature. However he cannot seem to access it. Can you help recover it to stop Evil Morty ?

- Didapatkan sebuah file .img
- Lalu saya menggunakan FTK Imager
- Lalu masukkan file tersebut di Image file diklik buka folder root
- 

- Didapatkan sebuah gambar yaitu flagnya
- 

- Dan didapatkan flag : **encryptCTF{alw4ys_d3lete_y0ur_f1les_c0mpletely}**
- Makasih Bang Maxtamvan atas pencerahannya :D

12. it's a WrEP – 50 point – Forensics –

Sniffed and Spoofed, but director called cut before final scene. Could you help Mr. Alderson to invade into eCorp?

Get the password for our Wifi Network "encryptCTF"

Submit flag as encryptCTF{</password/>}

- Didapatkan sebuah file .cap
- Lalu disini saya menggunakan aircrack-ng

#	BSSID	ESSID	Encryption
1	58:D7:59:79:61:34	MCSP	No data - WEP or WPA
2	40:31:3C:E6:CA:3C	Malik's	WPA (1 handshake)
3	14:CC:20:F5:32:FE	encryptCTF	WEP (88648 IVs)
4	0C:D2:B5:72:3C:D4	home	WPA (0 handshake)
5	E4:6F:13:80:82:99	prime2	No data - WEP or WPA
6	04:95:E6:05:50:60	Letzpay1	WPA (0 handshake)
7	C4:B8:B4:BF:5B:C8	foresightinn	No data - WEP or WPA
8	B4:EF:FA:51:EC:53	Le 2	No data - WEP or WPA
9	72:B7:AA:33:56:23	vivo 1802	None (0.0.0.0)
10	78:D3:8D:E4:E5:8C	Sarovar	None (172.16.168.188)

- Lalu terdapat encryptCTF di index ketiga WEP(88648 IVs)
- Lalu memulai melakukan PTW Attack
- Dan didapatkan flag : **encryptCTF{W45_17_R34L?!}**

13. Journey to the centre of the file 1 – 75 point

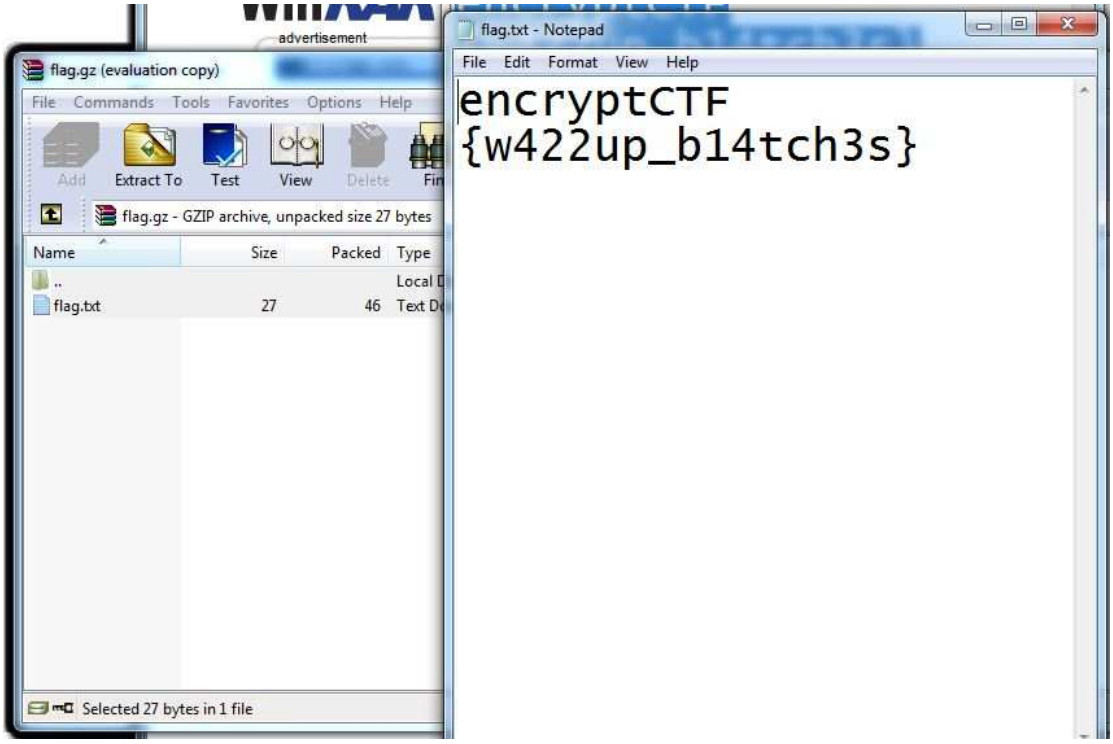
– Forensics –

"Nearly everything is really interesting if you go into it deeply enough …" - Richard Feynman

Author: maskofmydisguise

- Didapatkan sebuah file zip

- Lalu ketika di extract ternyata didalamnya masih ada zip
- Lalu ku extract semua secara manual



- Dan didapatkan flag : **encryptCTF{w422up_b14tch3s}**

14. Wi Will H4CK YOU!! – 100 point – Forensics –

Wifi security standards have been increased a lot in rencent times. But are they secure enough??? Get the password for our Wifi Network “encryptCTF”   Submit flag as encryptCTF{</password/>} captured.cap

- Didapatkan sebuah file .cap
- Lalu kulakukan bruteforce aircrack menggunakan wordlist rockyou.txt

```
aircrack-ng encryptCTFWPA.cap -w rockyou.txt
```

- Dan didapatkan flag : **encryptCTF{ThanckYou}**

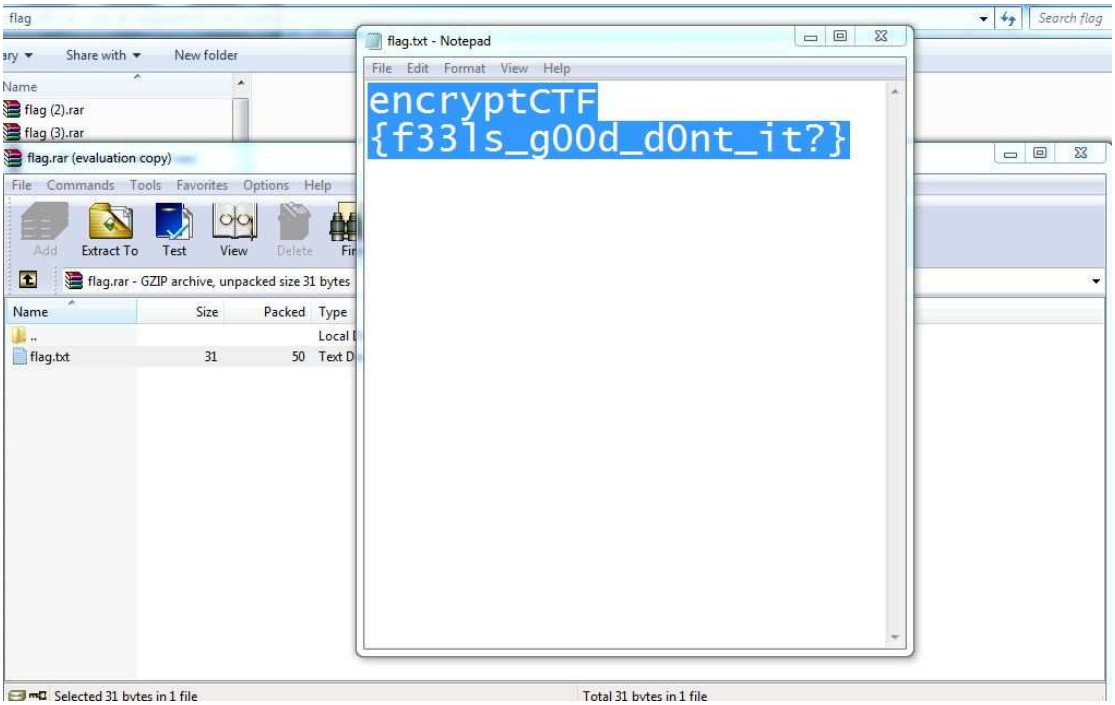
15. Journey to the centre of the file 2 – 150 point

– Forensics –

Improvise. Adapt. Overcome

Author: maskofmydisguise

- Didapatkan sebuah file zip
- Lalu ketika di extract ternyata didalamnya masih ada zip dan file tidak ada extensi tapi berupa zip juga
- Lalu ku extract dang anti extensi semua secara manual



- Dan didapatkan flag : **encryptCTF{f33ls_g00d_d0nt_it?}**

16. crackme01 – 75 point **– Reversing –**

this is crackme01. crackme01 is a crackme. so crackme! Author: @X3eRo0

- Didapatkan sebuah file ELF
- Lalu lihat string nya atau tarik aja ke notepad++
- Lalu search encrypt
- Dan didapatkan flag : **encryptCTF{gdb_or_r2?}**

17. Vault – 100 point **– Web –**

i heard you are good at breaking codes, can you crack this vault? http://104.154.106.182:9090 author: codacker

- Didapatkan sebuah web
- Lalu login dengan Sql Injection sederhana ' or '1'='1
- 

- Lalu check cookie dan terdapat base64
- Lalu decode to text
- Dan didapatkan flag : **encryptCTF{i_H4t3_inJ3c7i0n5}**