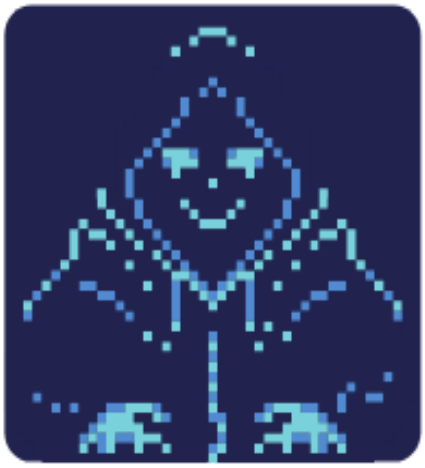
DOIT 5.0 CTF
WRITEUP



AVERAGE INTEL

ENJOYERS

exzettabyte (Paska Parahita)
banua (Muhammad Ichwan)
clonewriter (Ananda Fikri Ijlal Akbar)

UNIVERSITAS AMIKOM YOGYAKARTA

[DIGITAL FORENSIC]

Kasus Ann Dercover

Diberikan sebuah file pcap, pada deskripsi soal kita diminta untuk mencari alamat email “Ann Dercover”. Buka dengan wireshark, file pcap berisi traffic smtp, kami melakukan filter protocol smtp untuk memudahkan pencarian alamat email. Pada salah satu packet terdapat email dari Ann Dercover dengan alamat sneakyg33k@aol.com yang ditujukan ke sec558@gmail.com.

```
From: "Ann Dercover" <sneakyg33k@aol.com>
To: <sec558@gmail.com>
Subject: lunch next week
Date: Sat, 10 Oct 2009 07:35:30 -0600
MIME-Version: 1.0
Content-Type: multipart/alternative;
          boundary="----=_NextPart_000_0006_01CA497C.3E4B6020"
X-Priority: 3
X-MSMail-Priority: Normal
X-Mailer: Microsoft Outlook Express 6.00.2900.2180
X-MimeOLE: Produced By Microsoft MimeOLE V6.00.2900.2180
```

Flag : DOIT{sneakyg33k@aol.com}

Kasus Ann Dercover #2

Lanjutan dari kasus Ann Dercover yang pertama, file pcap menggunakan soal sebelumnya, pada deskripsi soal kita diminta untuk mencari password email Ann Dercover. Filter protocolnya smtp untuk memudahkan pencarian, pada salah satu packet terdapat *pass* yang terencode dalam base64 lakukan decode maka didapatkan 558r00lz.

```
User: c25lYWt5ZzMza0Bhb2wuY29t
334 UGFzc3dvcmQ6
Pass: NTU4cjAwbHo=
235 AUTHENTICATION SUCCESSFUL
MAIL FROM: <sneakyg33k@aol.com>
```

```
⇒  echo "NTU4cjAwbHo="|base64 -d
558r00lz%
```

Flag : DOIT{558r00lz}

Kasus Ann Dercover #3

Lanjutan dari kasus Ann Dercover yang kedua, file pcap menggunakan soal kasus Ann Dercover yang pertama, pada deskripsi soal kita diminta untuk mencari alamat email kekasih rahasia Ann Dercover. Pada salah satu packet berisi email dari Ann Dercover ke seseorang dimana isi email tersebut ada kata yang menandakan mempunyai hubungan lebih yaitu *sweetheart* dan *love*, serta Ann meminta untuk membawa *fake* passport dan bathing suit sepertinya Ann akan berenang bersama kekasihnya. Email kekasih rahasia Ann yaitu mistersecretx@aol.com

```
From: "Ann Dercover" <sneakyg33k@aol.com>
To: <mistersecretx@aol.com>
Subject: rendezvous
Date: Sat, 10 Oct 2009 07:38:10 -0600
MIME-Version: 1.0
Content-Type: multipart/mixed;
          boundary="----=_NextPart_000_000D_01CA497C.9DEC1E70"
X-Priority: 3
X-MSMail-Priority: Normal
X-Mailer: Microsoft Outlook Express 6.00.2900.2180
X-MimeOLE: Produced By Microsoft MimeOLE V6.00.2900.2180

This is a multi-part message in MIME format.

------=_NextPart_000_000D_01CA497C.9DEC1E70
Content-Type: multipart/alternative;
          boundary="----=_NextPart_001_000E_01CA497C.9DEC1E70"


------=_NextPart_001_000E_01CA497C.9DEC1E70
Content-Type: text/plain;
          charset="iso-8859-1"
Content-Transfer-Encoding: quoted-printable

Hi sweetheart! Bring your fake passport and a bathing suit. Address =
attached. love, Ann
```

Flag : DOIT{mistersecretx@aol.com}

Kasus Ann Dercover #4

Lanjutan dari kasus Ann Dercover yang ketiga, file pcap menggunakan soal kasus Ann Dercover yang pertama, pada deskripsi soal kita diminta untuk mencari tau nama lampiran yang dikirim Ann ke kekasih rahasianya, masih dengan packet yang sama dengan soal sebelumnya, scroll kebawah maka akan menemukan lampirannya yaitu secretrendezvous.docx.

```
Content-Type: application/octet-stream;
          name="secretrendezvous.docx"
Content-Transfer-Encoding: base64
Content-Disposition: attachment;
          filename="secretrendezvous.docx"
```

Flag : DOIT{secretrendezvous.docx}

[WEB EXPLOIT]

Aku ada dimana mana dan aku suka bawang!

Diberikan sebuah website “http://168.62.38.249/”. Pada tampilan utama terdapat button “bawang merah” yang berisi potongan flag pertama, kemudian pada page source terdapat flag part kedua, sedangkan pada file style.css terdapat flag part ketiga, dan pada file robots.txt terdapat flag part terakhir. Maka didapatkan flagnya, dan susun.


```
<!--# Flag Part 2 : "L13u_m4c_"-->

/*! # Flag Part 3 : "r!n990_"

# Final Flag Part 4 : "i<E<0k07}"
```

Flag : DOIT{R0&0t_L13u_m4c_r!n990_i<E<0k07}

Robots hiding! Base32+32

Diberikan sebuah website “http://168.62.38.249/robots/” namun tidak ada tampilan apa apa, sesuai judul challenge ini, kami mencurigai flag berada pada robots.txt. Langsung saja ke “http://168.62.38.249/robots/robots.txt” terlihat part dari semua flag dengan encoding base64. Copy semua part flag dan lakukan decode menggunakan bash didapatkan flag nya.

```
#Flag 1 = /UjBCMFRzLlR4VF8=

#flag 2 : /UHIzdmVOdF8=

# Final Flag Part 3 : /VzNCX0NSV0wzcg==
```

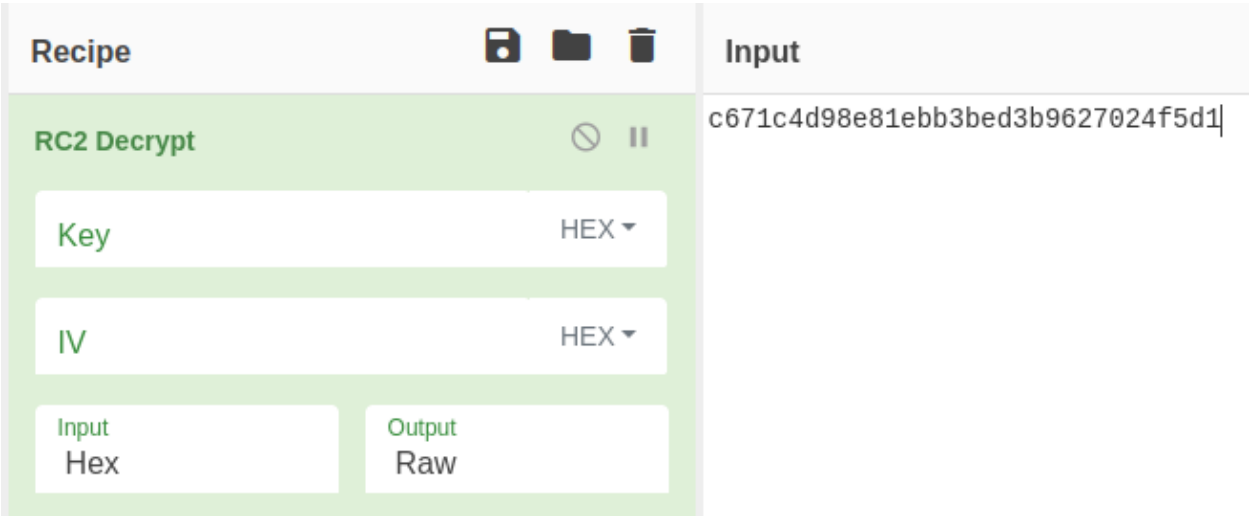
```
banua@basiber:~/Downloads/DOIT
$ echo "UjBCMFRzLlR4VF8=UHIzdmVOdF8=VzNCX0NSV0wzcg==" | base64 -d ; echo
R0B0Ts.TxT_Pr3veNt_W3B_CRWL3r
```

Flag : DOIT{R0B0Ts.TxT_Pr3veNt_W3B_CRWL3r}

Don’t need to exploit it! Something hidden here!

Diberikan sebuah website “http://168.62.38.249/somethinghere” kemudian cek page source pada website tersebut didapat potongan flag pertama dan flag kedua, dan flag pada tag href. Setelah dilihat lebih detail terdapat “RC2” dimana kami menduga bahwa potongan flag terakhir dienkripsi menggunakan RC2. lakukan decrypt menggunakan tools online pada CyberChef tanpa mengisi key dan iv, didapatkan potongan flag terakhir.

```
10      <h1>Now what?</h1>
11      <!-- u expose me  : {5point0 -->
12      <p>you're here now what?</p>
13      <p style="color:white">_G3T_s0L1d_w</p>
14      <a href="flag:c671c4d98e81ebb3bed3b9627024f5d1" target="_blank" rel="noopener noreferrer">RC2</a>
```



Output

```
1th_c0mpT1Tiv3}
```

Flag : DOIT{5point0_G3T_s0L1d_w1th_c0mpT1Tiv3}

Java anu lho~

Diberikan sebuah website “http://168.62.38.249/sometxt/”. Pada tampilan utama website tersebut hanya ada text biasa. Cek pada page source didapat code javascript yang telah di obfuscate. Copy code tersebut kemudian buka console pada website dan paste pada console maka didapatkan flag nya.

```
» 'use strict';var _0x17dac3=_0x305b;(function(_0x42db9a,_0x3e368e){var _0x229367
  (parseInt(_0x229367(0x125))/0x2)+parseInt(_0x229367(0x12f))/0x3+-parseInt(_0x22
  /0x6*(-parseInt(_0x229367(0x12d))/0x7)+-parseInt(_0x229367(0x12c))/0x8*(parseIr
  if(_0x592a5c===_0x3e368e)break;else _0xead6c0['push'](_0xead6c0['shift']());}ca
  foo=['5_s4f3}','0bfus'];function _0x305b(_0x2fcd8d,_0x3c4895){var _0x1e3da7=_0x
  _0x3a2f59=_0x1e3da7[_0x305b36];return _0x3a2f59;},_0x305b(_0x2fcd8d,_0x3c4895);
  ['push','972786eQgKvD','927YkMAUL','18014URGnet','10EmnCcS','_y0ur_j','0x1','16
  0ZXwxoT'];_0x1e3d=function(){return _0x2896e3;};return _0x1e3d();}(function(_0x
  {_0x49ec74[_0x4ea2b5(0x122)](_0x49ec74['shift']());}};_0x53f49d(++_0x368e67);}(
  _0x95ff18=foo[_0x3345db];return
  _0x95ff18;},fOo=Foo,k12po5j1o2j5ojqnfqwnrkqwrnkqwrbk1j2h51hkjrk12h512h512p51i25
  i125hui15hi1uh5iu=_0x17dac3(0x130);
  k12po5j1o2j5ojqnfqwnrkqwrnkqwrbk1j2h51hkjrk12h512h512p51i25p1o2ipo1ipo2andansko
  2po5j1o2j5ojqnfqwnrkqwrnkqwrbk1j2h51hkjrk12h512h512p51i25p1o2ipo1ipo2andanskdna
  _0x17dac3(0x12b)),k12po5j1o2j5ojqnfqwnrkqwrnkqwrbk1j2h51hkjrk12h512h512p51i25p1
  25hui15hi1uh5iu=k12po5j1o2j5ojqnfqwnrkqwrnkqwrbk1j2h51hkjrk12h512h512p51i25p1o2
  hui15hi1uh5iu+'c4t3_m4k3',k12po5j1o2j5ojqnfqwnrkqwrnkqwrbk1j2h51hkjrk12h512h512
  hu15hui125hui15hi1uh5iu=k12po5j1o2j5ojqnfqwnrkqwrnkqwrbk1j2h51hkjrk12h512h512p5
  15hui125hui15hi1uh5iu+_0x17dac3(0x127),k12po5j1o2j5ojqnfqwnrkqwrnkqwrbk1j2h51hk
  4ui1hiuh5i1u5hu15hui125hui15hi1uh5iu=k12po5j1o2j5ojqnfqwnrkqwrnkqwrbk1j2h51hkjr
  i1hiuh5i1u5hu15hui125hui15hi1uh5iu+fOo(_0x17dac3(0x128));
← "DOIT{0bfusc4t3_m4k3_y0ur_j5_s4f3}"
```

Flag : DOIT{0bfusc4t3_m4k3_y0ur_j5_s4f3}

[STEGANOGRAPHY]

Don’t Overthinking!

Diberikan sebuah file image “gembok-onlen.jpg”, download file tersebut kemudian cek metadata menggunakan “exiftool”, terlihat hexadecimal di bagian Copyright dimana sesuai dengan deskripsi 16 yang artinya base 16 (hexadecimal). Copy dan decode menggunakan bash didapatkan flagnya.

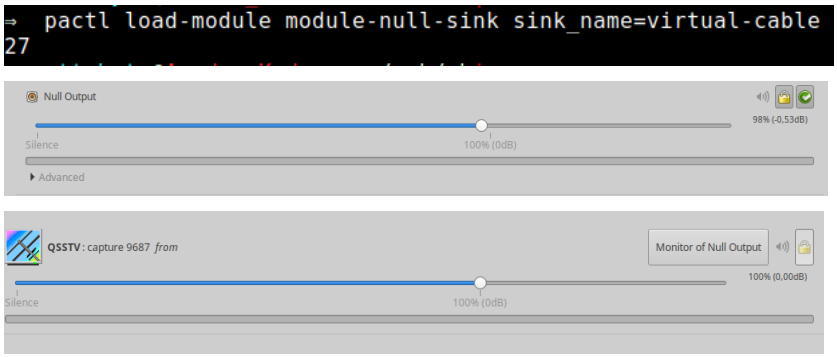
```
$ exiftool gembok-onlen.jpg
ExifTool Version Number         : 10.80
File Name                       : gembok-onlen.jpg
Directory                       : .
File Size                       : 691 kB
File Modification Date/Time     : 2021:10:13 13:58:12+07:00
File Access Date/Time           : 2021:10:13 13:58:22+07:00
File Inode Change Date/Time     : 2021:10:13 13:58:20+07:00
File Permissions                : rw-rw-r--
File Type                       : JPEG
File Type Extension             : jpg
MIME Type                       : image/jpeg
JFIF Version                    : 1.01
Exif Byte Order                 : Big-endian (Motorola, MM)
Image Description               : 3f8de088ad769a6e3c7abc45d576cdb3
Orientation                     : Horizontal (normal)
X Resolution                    : 72
Y Resolution                    : 72
Resolution Unit                 : inches
Software                        : GIMP 2.8.2
Modify Date                     : 2014:04:30 16:31:59
Copyright                       : 45 5a 5f 33 78 31 46 5f 74 30 6f 4c
```

```
banua@basiber:~/Downloads/DOIT
$ exiftool gembok-onlen.jpg | grep Copyright
Copyright                       : 45 5a 5f 33 78 31 46 5f 74 30 6f 4c
banua@basiber:~/Downloads/DOIT
$ echo 45 5a 5f 33 78 31 46 5f 74 30 6f 4c | xxd -r -p ; echo
EZ_3x1F_t0oL
banua@basiber:~/Downloads/DOIT
```

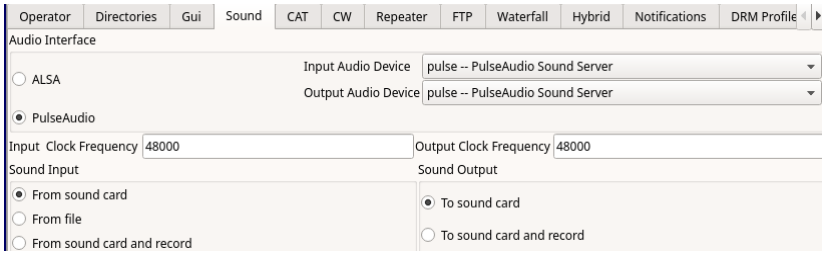
Flag : DOIT{EZ_3x1F_t0oL}

LapanA3/ORARI

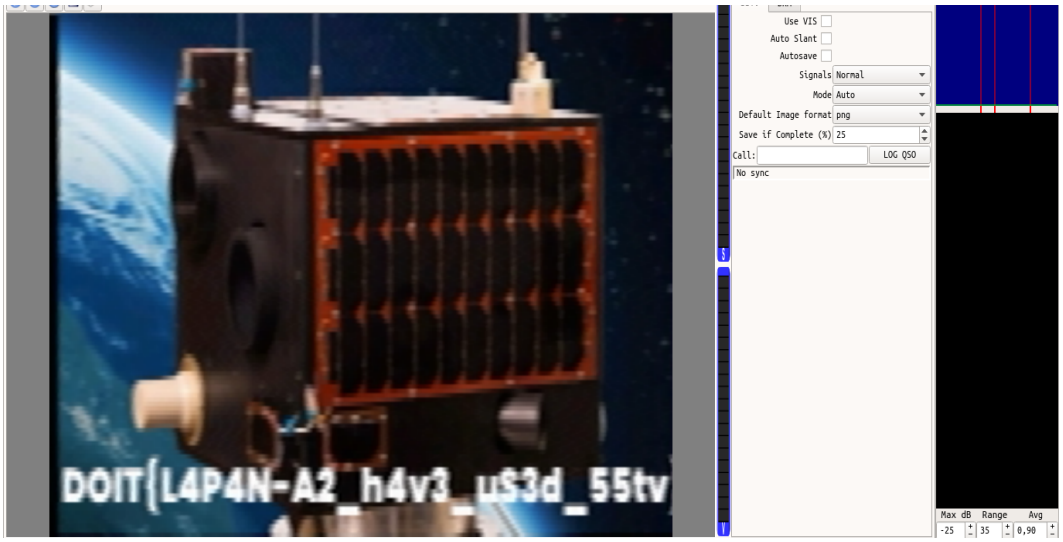
Diberikan file wav, ketika diplay audio tersebut khas dari sstv (slow-scan television), dapat didecode dengan qsstv tapi sebelumnya konfigurasi module null output dengan pactl bertujuan agar nantinya audio yang diplay tidak menjadi output melainkan menjadi input untuk qsstv. Jika sudah ada null output pada pavucontrol maka sudah aktif, lalu arahkan qsstv untuk record dari null output.



Konfigurasi juga pada qsstv agar menggunakan audio interface pulseaudio



Lalu play dan didapatkan flag



Flag : DOIT{L4P4N-A2_h4v3_uS3d_55tv}

Onodera has stolen my key!

Diberikan file jpg yang ketika dilakukan binwalk ditemukan file lain, extract dengan *binwalk -e.*

```
⇒  binwalk onodera.jpg

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, JFIF standard 1.01
42451         0xA5D3          POSIX tar archive (GNU), owner user name: "zip"
```

Hasil extract dengan binwalk didapat 2 file yaitu ingpo.txt dan Onodera.zip, file zip tersebut perlu password untuk extract.

```
⇒  binwalk -e onodera.jpg

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, JFIF standard 1.01
42451         0xA5D3          POSIX tar archive (GNU), owner user name: "zip"

exzettabyte@Avadra_Kedavra:~/Documents/doit|
⇒  ls
a.pcap  flag  LapanA2.wav  onodera.jpg  _onodera.jpg.extracted
exzettabyte@Avadra_Kedavra:~/Documents/doit|
⇒  ls _onodera.jpg.extracted
A5D3.tar  ingpo.txt  Onodera.zip
```

Pada ingpo.txt ditemukan base64 yang sepertinya usefull, decode base64nya lalu extract zipnya dengan password yang didapat.

```
m4yb3
us3full
:aU4xUDQ1NWFOdU55NAo=
```

```
⇒  echo "aU4xUDQ1NWFOdU55NAo="|base64 -d
iN1P455aNuNy4
```

Didapat flag yang dipotong menjadi 5 file jpg

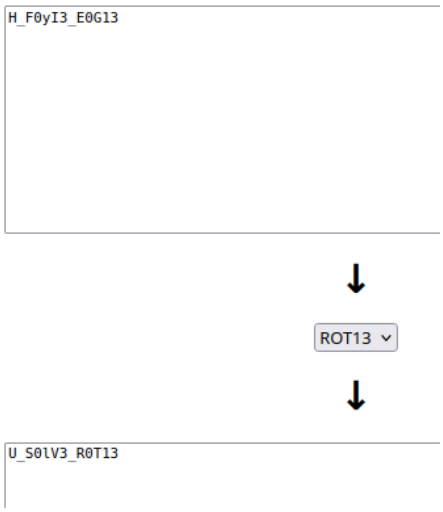
```
⇒  ls
 A5D3.tar  'flag(1).jpg'  'flag(2).jpg'  'flag(3).jpg'  'flag(4).jpg'   flag.jpg   ingpo.txt   Onodera.zip
```

Flag : DOIT{PLEASE!_G3T_B4CK_MY_K3Y}

[CRYPTOGRAPHY]

Heavy Rotation

Diberikan sebuah deskripsi dimana terdapat text yang sepertinya telah dienkripsi. Namun pada deskripsi tersebut seakan memberikan hint dimana terdapat ROT13. Kami menduga enkripsi yang digunakan adalah ROT13. Langsung saja kami decrypt menggunakan tools online “https://rot13.com” didapatkan flagnya.

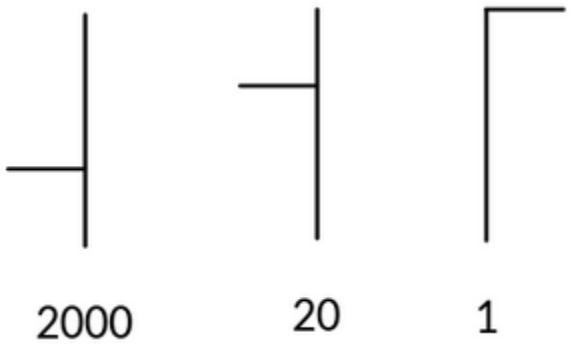


Flag : DOIT{U_S0lV3_R0T13}

Biarawan

Diberikan sebuah deskripsi yang hanya berisi sebuah simbol. Pada judul soal kalau ditranslate kedalam bahasa inggris artinya adalah monks. Kami menduga flag adalah plaintext dari monks cipher tersebut. Browsing di google dan lakukan analisa apa itu monks dan bagaimana algoritmanya pada “https://en.wikipedia.org/wiki/The_Ciphers_of_the_Monks” memberikan kesimpulan bahwa flag dimulai dari ribuan => puluhan => satuan. Maka didapatkan flag nya.
Flag simbol :

Satuan flag yang harus diterjemahkan :



Flag : DOIT{2021}

Bary Sylla

Diberikan sebuah deskripsi dengan chipertext dari flag nya ([00] [92] [65] [42] [83] [87] [41] [14]). Pertama kami menduga itu adalah phone keypad chiper namun ternyata salah. Melihat kembali ke judul soal sepertinya ini adalah Syllabary Cipher. Langsung saja browsing di google, kami menemukan website "https://sites.google.com/site/bionspot/the-syllabary-cipher" dengan basic key-square. Namun dekripsi yang kami lakukan masih salah. Lalu coba browsing website lainnya untuk mencari key-square yang lain dan menemukan "https://asecuritysite.com/challenges/sy" dengan key-square yang berbeda dengan sebelumnya. Lakukan dekripsi menggunakan format row-column (misal [92] maka row (baris) 9 dan column (kolom) 2 maka plain textnya WE), dan didapatkanlah flagnya.

	6	7	1	9	4	3	2	5	0	8
8	C	3	H	8	AR	M	ING	P	RI	N
5	CE	A	1	AL	AN	AND	ARE	AS	AT	ATE
0	ATI	B	2	BE	CA	CO	COM	D	4	DA
2	DE	E	5	EA	ED	EN	ENT	ER	ERE	ERS
3	ES	EST	F	6	G	7	HAS	HE	I	9
4	IN	ION	IS	IT	IVE	J	Ø	K	L	LA
1	LE	ME	ND	NE	NT	O	OF	ON	OR	OU
6	Q	R	RA	RE	RED	RES	RO	S	SE	SH
7	ST	STO	T	TE	TED	TER	TH	THE	THI	THR
9	TI	TO	U	V	VE	W	WE	X	Y	Z

CT : [00] = 4,
[92] = WE,
[65] = S,
[42] = Ø,
[83] = M,

[87] = 3,
[41] = IS,
[14] = NT

Flag : DOIT{4WESØM3ISNT}

[MISCELLANEOUS]

Check Your Sanity!

Flag terdapat pada deskripsi soal

Check Your Sanity!

10

Hi, welcome to CTF** DoIT 5.0!**

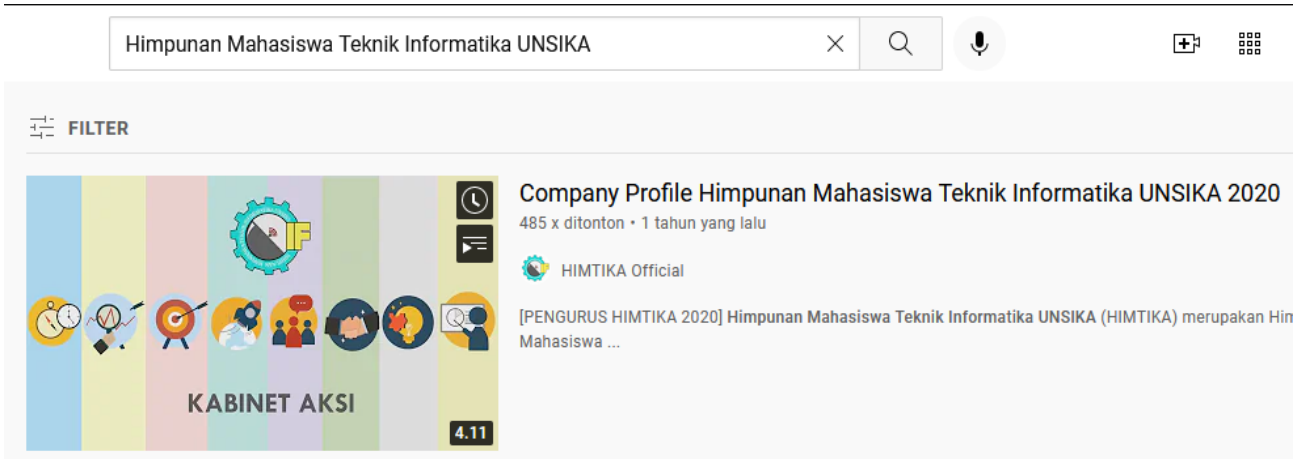
Flag format: **DOIT{flag}**

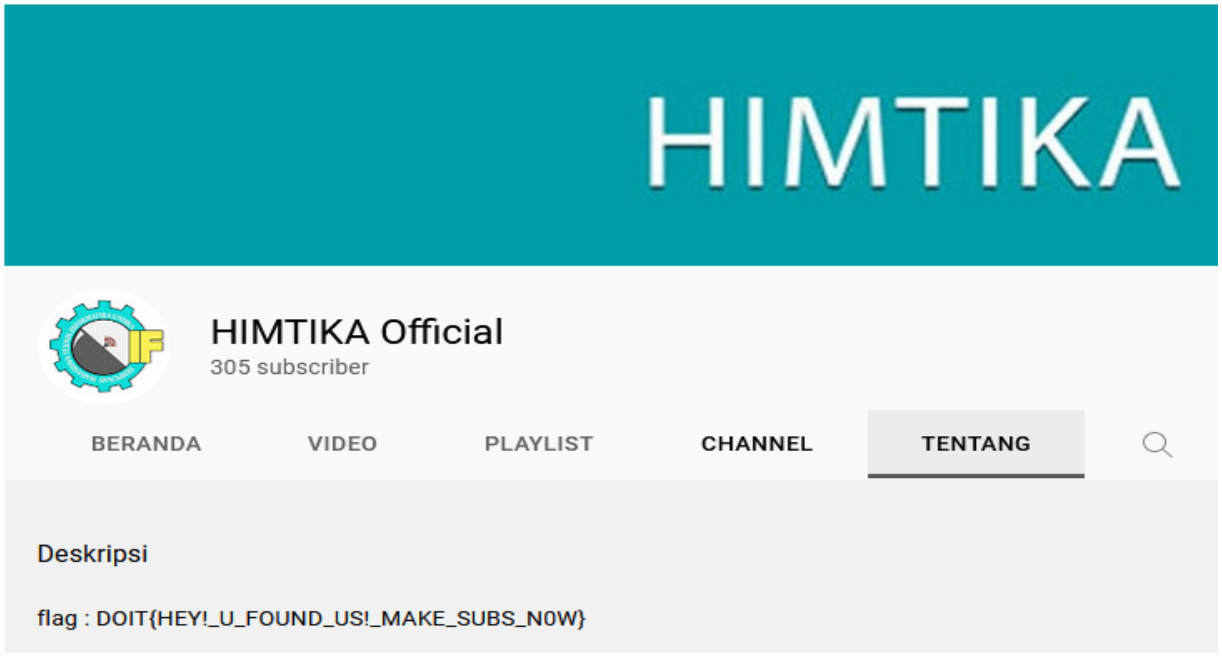
flag: DOIT{w3lc0me_t0_DoIT_5.0}

Flag : DOIT{w3lc0me_t0_DoIT_5.0}

Find us on Youtube!

Diberikan sebuah deskripsi yaitu “Himpunan Mahasiswa Teknik Informatika UNSIKA”. Sesuai judul soal dimana kita harus menemukan channel tersebut di youtube. Lakukan pencarian dengan keyword “Himpunan Mahasiswa Teknik Informatika UNSIKA” di youtube maka didapatkan channel HIMTIKA Official, buka channel tersebut dan flag berada pada bagian “Tentang”.





Flag : DOIT{HEY!_U_FOUND_US!_MAKE_SUBS_N0W}