FORENSICS 🔍

What is it?

- Looking for digital “clues”
  - Data hiding in audio, video, images
  - Encoded data
  - Deleted or corrupted data

- It is a profession

- There are lots of videos of people talking about catching people hiding info, which are funny.

Here’s one: DEFCON 21 - Forensic Fails

https://www.youtube.com/watch?v=NG9Cg_vBKOg

Plan

- Briefly discuss tools and methods
- Give a demo (from DEFCON)
- Try some challenges

Image Steganography

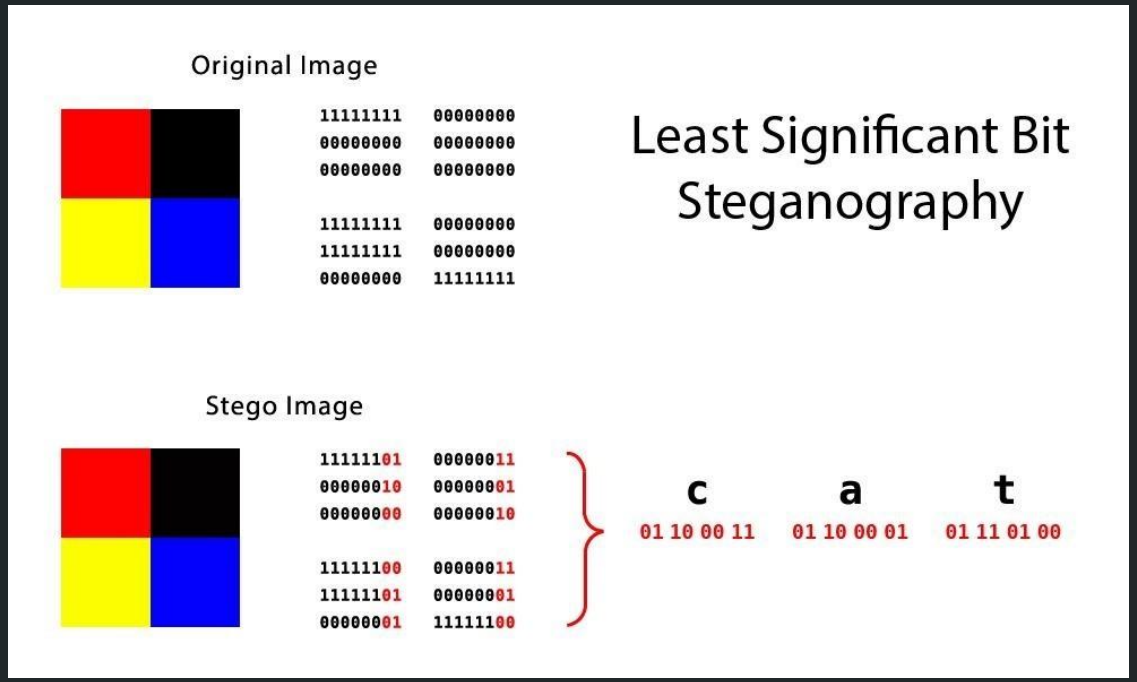
- Hiding data in images
- Techniques:
  - Hiding data in metadata
  - LSB
  - Hiding data in the image source
  - Encrypting data and “mixing” it into the image
  - Hiding data in the image itself

Hiding data in the metadata:

- Metadata - gives basic information about the data (ie. its size, its location, creation time, etc)
- Sometimes data (by data I mean flags) is in the Comments of the metadata
- To read metadata: exiftool [FILENAME]
- You can also use exiftool to write to the metadata
- Ex:
-

```
crazyeights@kali: ~/Downloads/ctf_primer_01/forensics
crazyeights@kali:~/Downloads/ctf_primer_01/forensics$ exiftool -a f.02.wav
ExifTool Version Number         : 11.80
File Name                       : f.02.wav
Directory                       : .
File Size                       : 394 kB
File Modification Date/Time     : 2019:10:26 13:57:46-04:00
File Access Date/Time           : 2020:02:02 19:41:43-05:00
File Inode Change Date/Time     : 2020:02:02 19:41:41-05:00
File Permissions                : rw-rw-r--
File Type                       : WAV
File Type Extension             : wav
MIME Type                       : audio/x-wav
```

LSB (Least Significant Bit):



More:
https://www.cybrary.it/0p3n/hide-secret-message-inside-image-using-lsb-steganography/

Hiding data in the image source

Example:

```
sphil@athens:~/Documents/Screenshots/steg images$ hexdump anon.jpg | tail -10
*
000c150 28 00 a2 8a 28 00 a2 8a 28 00 a2 8a 28 00 a7 0f
000c160 ba 69 b4 e1 d0 d0 03 68 a2 8a 00 28 a2 8a 00 28
000c170 a2 8a 00 28 a2 8a 00 28 a2 8a 00 28 a2 8a 00 28
*
000c1a0 a2 8a 00 28 a2 8a 00 2b fa e5 ff 00 82 61 ff 00
000c1b0 c9 8d fc 35 ff 00 b8 cf fe 9e 2f 6b f9 1a af eb
000c1c0 97 fe 09 87 ff 00 26 37 f0 d7 fe e3 3f fa 78 bd
000c1d0 a0 0f ff d9
000c1d4
sphil@athens:~/Documents/Screenshots/steg images$ echo 'hello world' >> anon.jpg
sphil@athens:~/Documents/Screenshots/steg images$ hexdump anon.jpg | tail -10
*
000c150 28 00 a2 8a 28 00 a2 8a 28 00 a2 8a 28 00 a7 0f
000c160 ba 69 b4 e1 d0 d0 03 68 a2 8a 00 28 a2 8a 00 28
000c170 a2 8a 00 28 a2 8a 00 28 a2 8a 00 28 a2 8a 00 28
*
000c1a0 a2 8a 00 28 a2 8a 00 2b fa e5 ff 00 82 61 ff 00
000c1b0 c9 8d fc 35 ff 00 b8 cf fe 9e 2f 6b f9 1a af eb
000c1c0 97 fe 09 87 ff 00 26 37 f0 d7 fe e3 3f fa 78 bd
000c1d0 a0 0f ff d9 68 65 6c 6c 6f 20 77 6f 72 6c 64 0a
000c1e0
sphil@athens:~/Documents/Screenshots/steg images$
```

Using steghide, stegosuite, ...

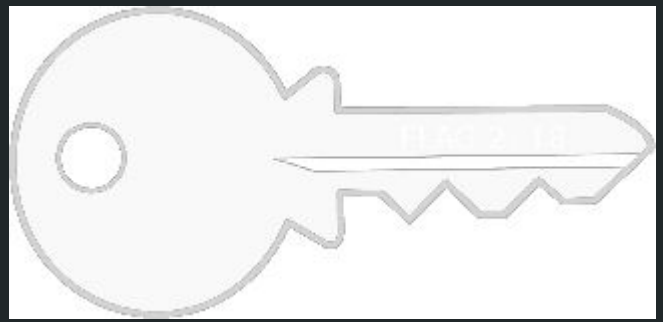
Steghide and stegosuite let you encrypt files and hide them inside another files content
- Limitations:
    - The file used for attaching the hidden files (cover files) are limited to certain formats (JPEG, BMP, ...)
- The cover file must be bigger than the file you want to hide

According to its manual, this is what it does.

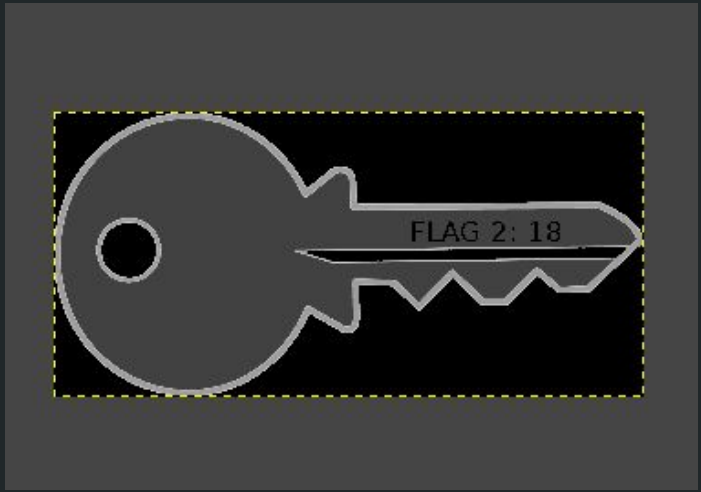
If you hide a text file using an image, then after ‘Steghide’ encrypts and compresses the text file, it’ll also ‘convert’ its data to look like it holds data of an image (pixels!). If you use an audio file instead of the image, then it’ll ‘convert’ the text data into ‘audio samples’. That’s how the data becomes ‘invisible’ once gets mixed.

Hiding data in the image itself

- Used less frequently
- Text is hidden through a series of filters to the image
- Reversing the filters reveals to text

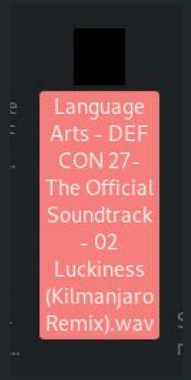


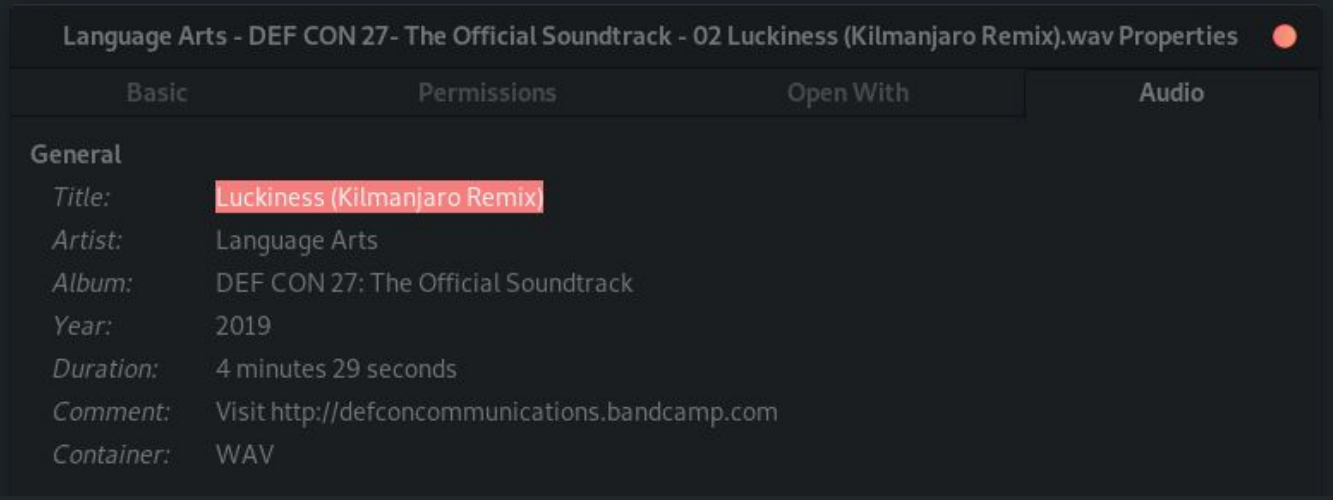
Linear Invert



A quick example:

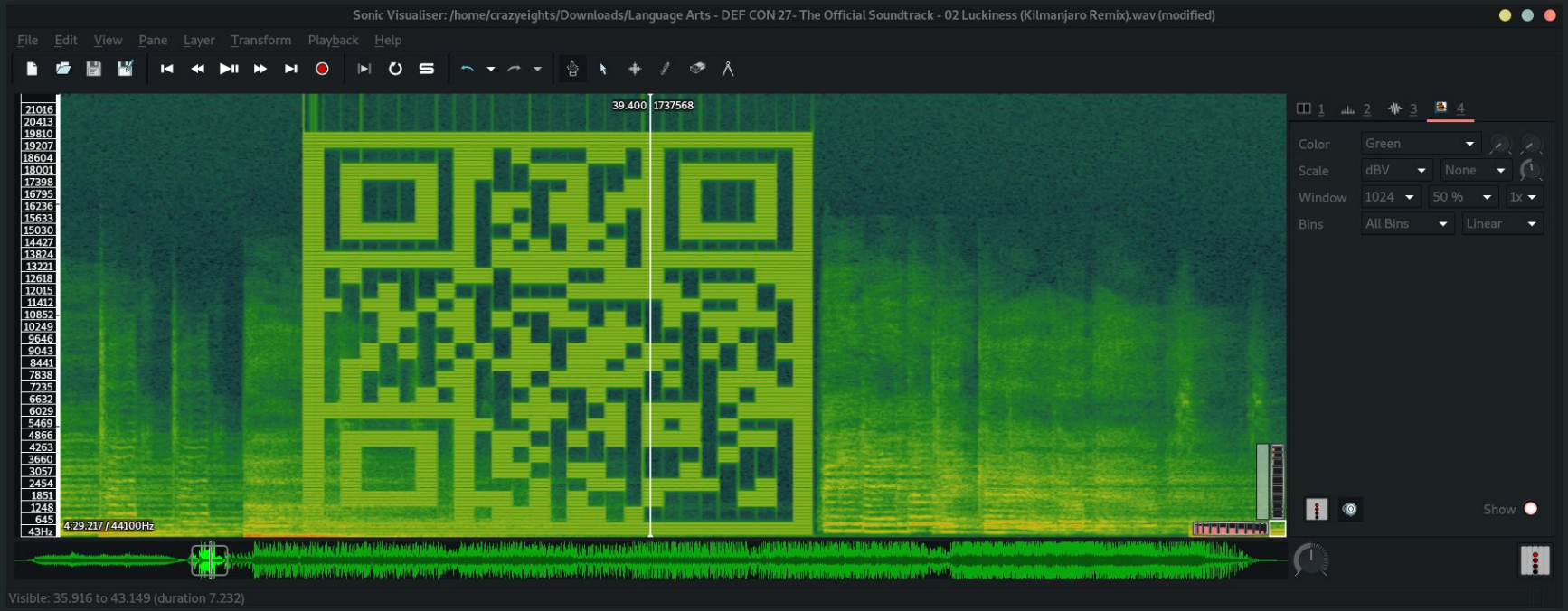
Find the flag given this audio file. The music actually quite pleasant :)





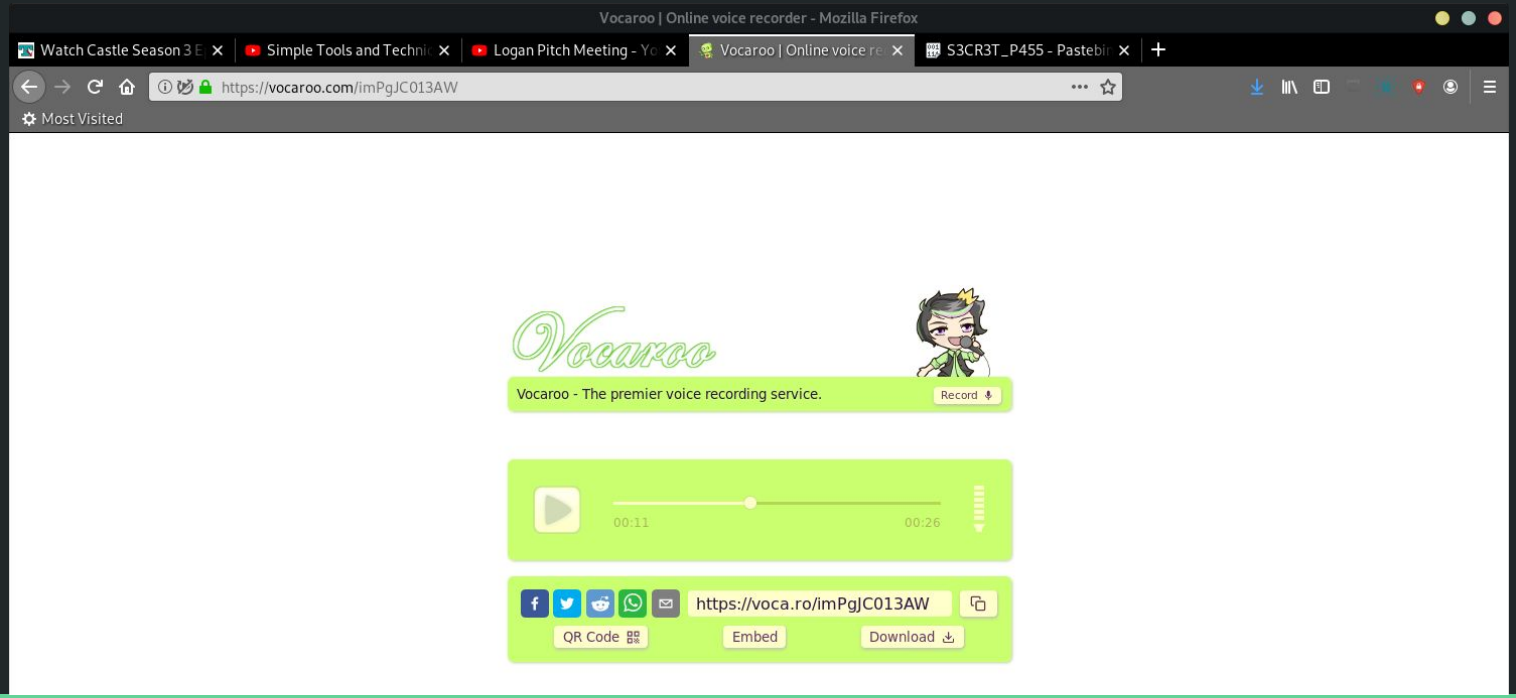
A quick example:

Using Sonic Visualiser open the song, and add stegrogram layer:



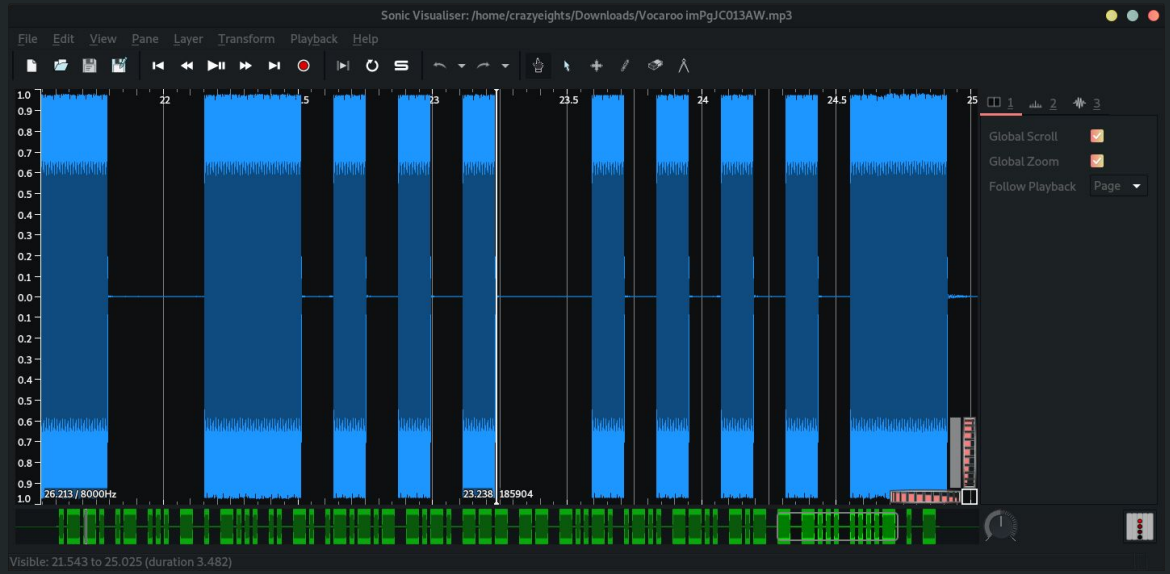
A quick example:

Used a QR reader to get the corresponding url



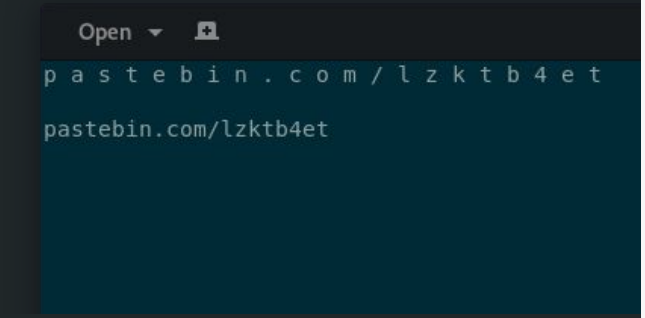
A quick example:

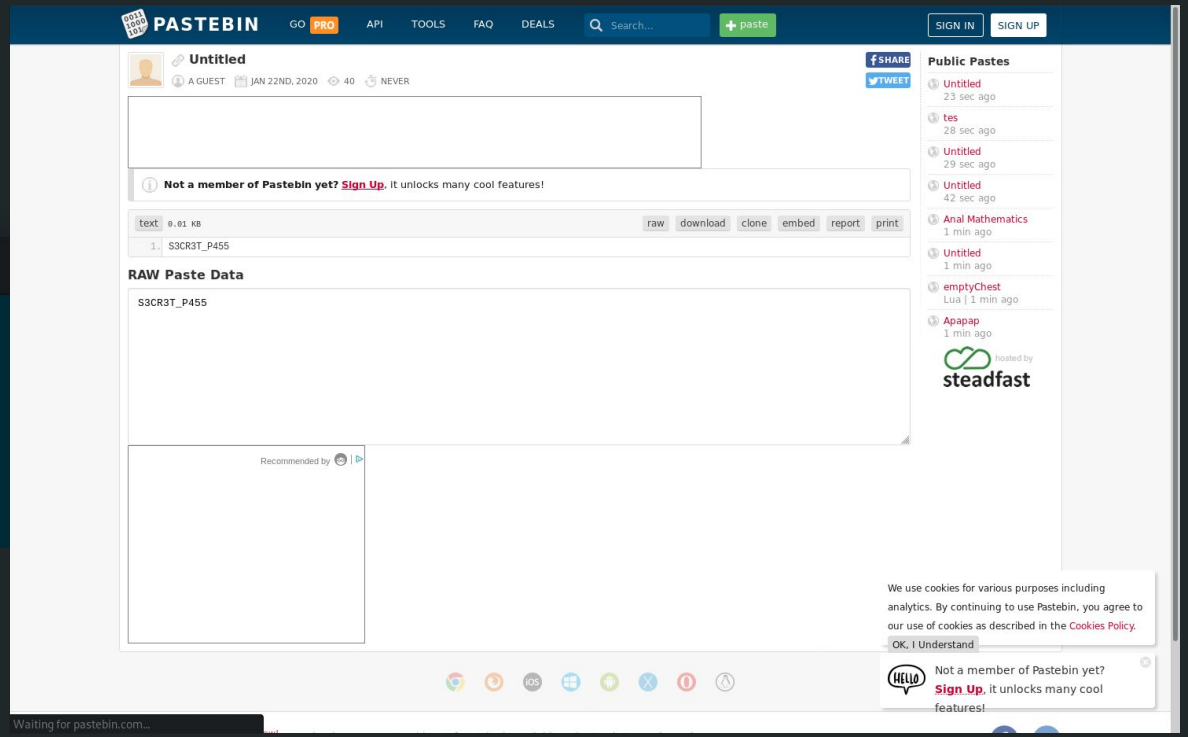
Listened to it and it just some beeps. Its morse code. Since I am not fast enough to translate just by listening I downloaded it and analysed the wavelengths: (You can see that dots are shorter and dashes are longer)



A quick example:

I got:





A quick example:

Use the password to retrieve the flag from the image:

```
crazyeights@kali: ~/Downloads
crazyeights@kali:~/Downloads$ steghide extract -sf "Language Arts - DEF CON 27- The Official Soundtrack - 02 Luckiness (Kilmanjaro Remix).wav" -p S3CR3T_P455
wrote extracted data to "secret".
crazyeights@kali:~/Downloads$
```

```
crazyeights@kali: ~/Downloads
crazyeights@kali:~/Downloads$ cat secret
THM{f0und_m3!}
crazyeights@kali:~/Downloads$
```

Tools Needed:

- exiftool
- steghide
- binwalk
- wireshark
- sonic visualizer
- zsteg
- stegoveritas
- Hex editor (I use ghex)
- CYBERCHEF (or other online tools)
- https://www.dcode.fr => Has many different decoders
- https://copy.sh/brainfuck/ ⇒ Brainf*ck compiler

Types of codes:

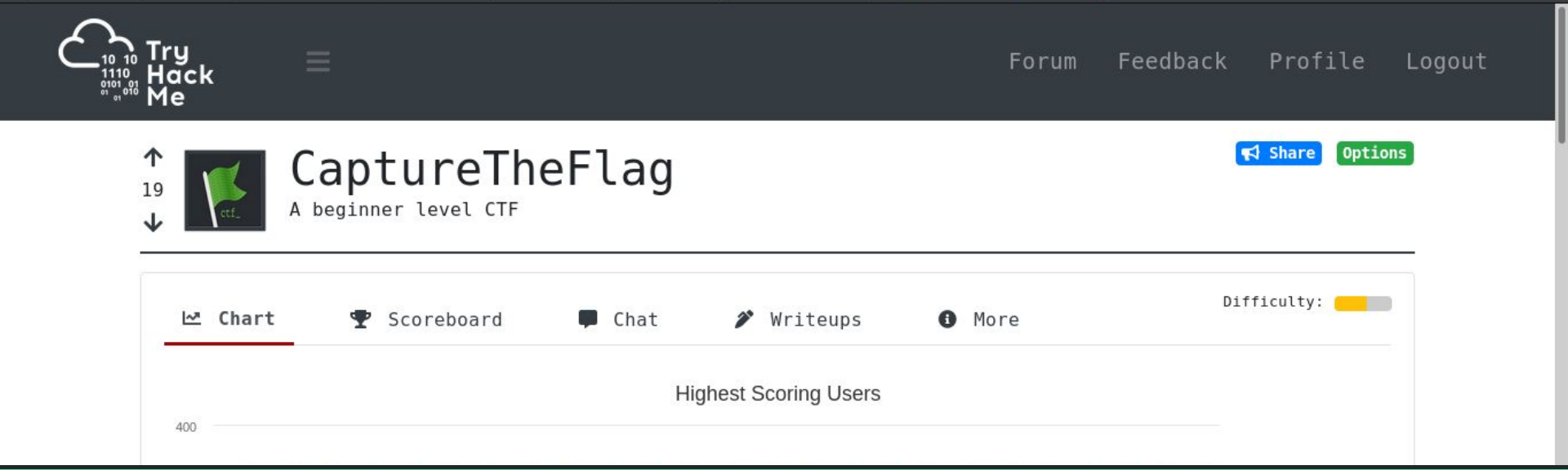
- Different Data Formats: base85, base64, base62, base32, base16
    - Ascii is base10
- Different Encoding types: ROT13 (CAESAR CIPHER), ROT47, VIGENERE CIPHER
    - Characters are rotated, ie. ROT3 A ⇒ D
- XOR cipher, morse code
- Different Languages: Aurebesh, Pigpen Cipher, Brainf*ck

Challenges, ya let’s go! 🏆👽

CTF: functf: this was retired now. :(

https://www.tryhackme.com/room/functf

No SSH Connection Required



In my github:

https://github.com/crazyeights225/CCSC/tree/master/functf-retired

Challenges in own folder with challenge description

Flag format: tryhackme{...}

[Part 1] #1 Do Images have strings?

#1 Do Images have strings?

The hint here is strings

strings - print the sequences of printable characters in files

root@kali:~/Downloads# strings Basic.jpg

JFIF
ICC_PROFILE
...

tryhackme{7h1s_i5_wh4t_strings_d0es} ⇐ ANSWER TO #1

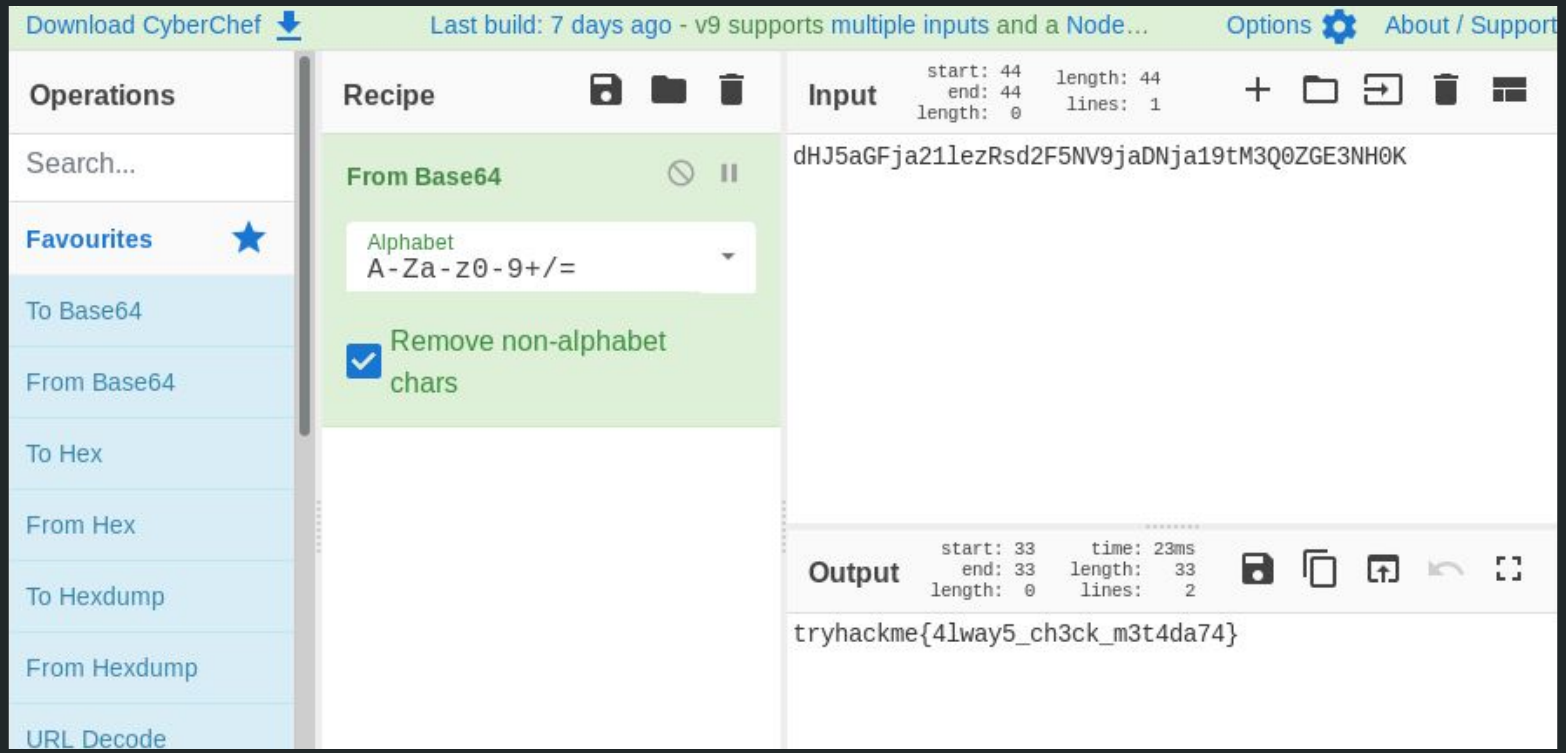
[Part 1] #2 Metadata or EXIF data?.....ah!! I'm so confused

#2 Metadata or EXIF data?.....ah!! I'm so confused

➜ Metadata or Exif data can be viewed with exiftool

```
root@kali:~/Downloads# exiftool Basic.jpg

...

Comment                         : dHJ5aGFja21lezRsd2F5NV9jaDNja19tM3Q0ZGE3NH0K

Image Width                     : 404

Image Height                    : 404

Encoding Process                : Progressive DCT, Huffman coding
```

#2 Metadata or EXIF data?.....ah!! I'm so confused



[Part 2] #1 Find the flag.

#1 Find the flag.

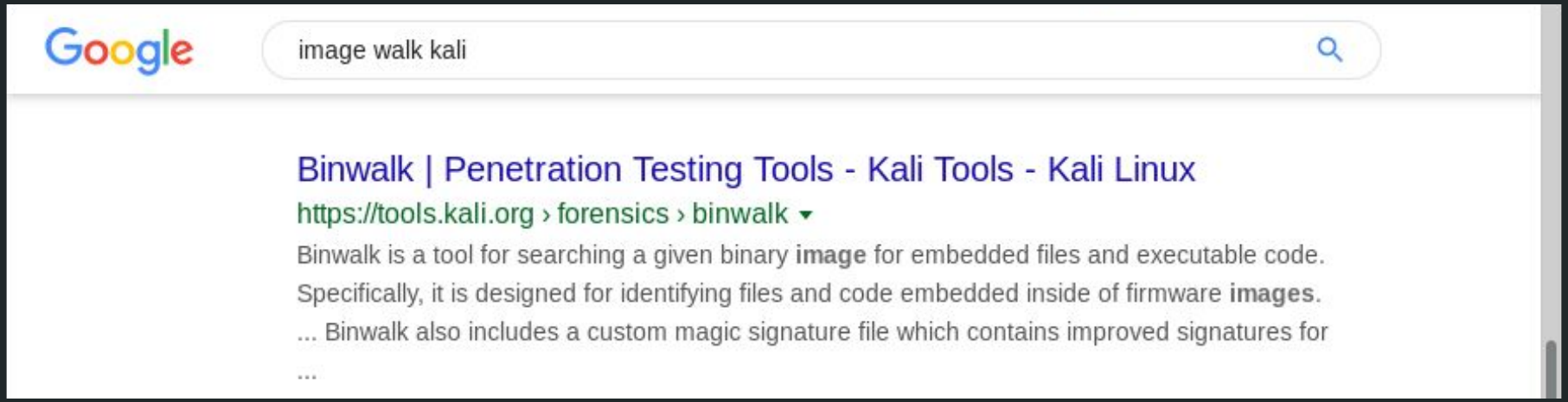
Download the next image: walk.jpg

I can make this easy just by telling you the tool or maybe you can read the title again and figure out your self.

P.S - It's a very famous, open source tool :)

#1 Find the flag.

The image name is walk, so:



The tool is binwalk

#1 Find the flag.

binwalk:

binwalk - tool for searching binary images for embedded files and executable code

Param:

-e, --extract Automatically extract known file types

#1 Find the flag.

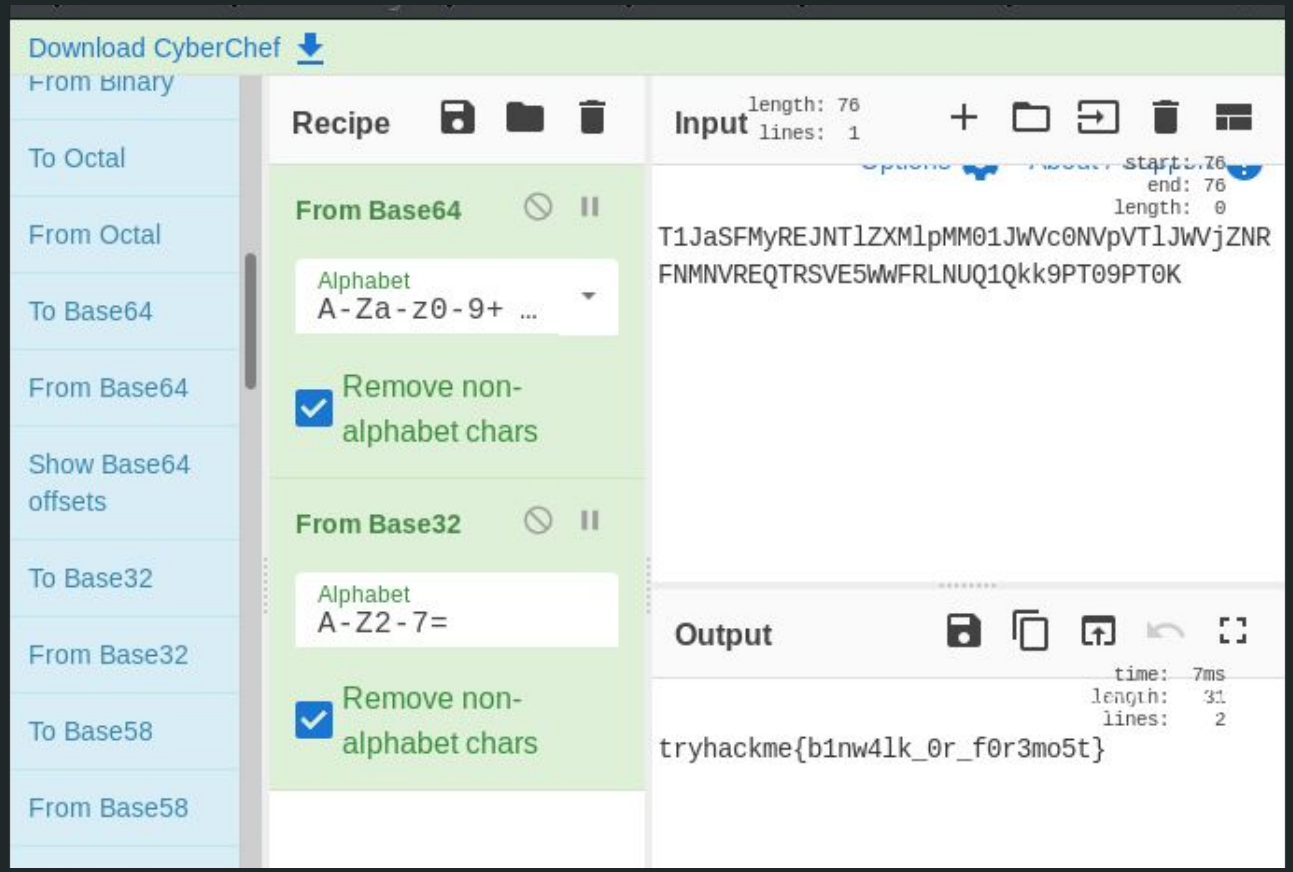
```
root@kali:~/Downloads# binwalk -e walk.jpg

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, JFIF standard 1.01
30            0x1E            TIFF image data, big-endian, offset of first image directory: 8
170610        0x29A72         gzip compressed data, from Unix, last modified:
2019-04-21 08:25:56
root@kali:~/Downloads#ls

 _walk.jpg.extracted
```

#1 Find the flag.

```
root@kali:~/Downloads# cd _walk.jpg.extracted/
root@kali:~/Downloads/_walk.jpg.extracted# ls
29A72  29A72.gz
root@kali:~/Downloads/_walk.jpg.extracted# cat 29A72
PaxHeader/flag.txt000644 001750 001751 00000000066 13457024252 014506
xustar00mzfrmzfr000000 000000 30 mtime=1555835050.729934811
24 SCHILY.fflags=extent
flag.txt000644 001750 001751 00000000352 13457024252 012533
0ustar00mzfrmzfr000000 000000 hmm..So you've got the flag.txt file good!!
Now let's play a bit with bases
This is the flag but it's encoded twice with 2 different bases. Figure it out

T1JaSFMyREJNTlZXMlpMM01JWVc0NVpVTlJWVjZNRFNMNVREQTRSVE5WWFRL
NUQ1Qkk9PT09PT0K
```

#1 Find the flag.

I love cyberchef



[Part 3] #1 Find the Flag

#1 Find the flag.

Download the next image: hide.jpg

Hint: You know the drill, focus on the Title.

#1 Find the flag.

This tool is really popular:

steghide - a steganography program

To extract:

Example:

$ steghide extract -sf picture.jpg

Enter passphrase:

wrote extracted data to "secret.txt".

#1 Find the flag.

```
root@kali:~/Downloads# steghide extract -sf hide.jpg

Enter passphrase:

steghide: could not extract any data with that passphrase!

root@kali:~/Downloads#
```

Oh No. The passphase must be hidden in the image.

#1 Find the flag.

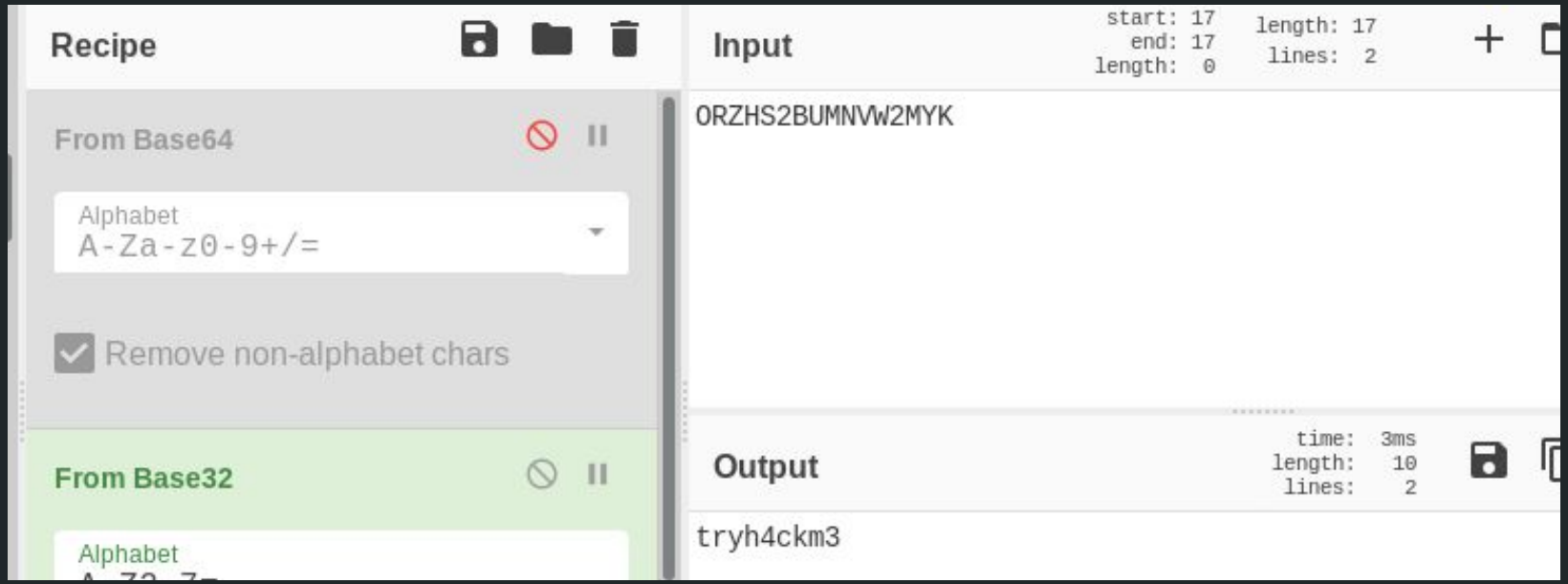
You can find the password 2 ways:

```
root@kali:~/Downloads# strings hide.jpg
JFIF
ORZHS2BUMNVW2MYK
$3br
%&'()*456789:CDEFGHIJSTUVWXYZcdefghijstuvwxyz
        #3R
&'()*56789:CDEFGHIJSTUVWXYZcdefghijstuvwxyz
H@9l|
3_`x
NM+U
V[$2\
```

```
EAuy-
root@kali:~/Downloads# exiftool hide.jpg
ExifTool Version Number         : 11.77
File Name                       : hide.jpg
Directory                       : .
File Size                       : 56 kB
File Modification Date/Time     : 2019:12:27 19:23:3
File Access Date/Time           : 2019:12:27 19:28:4
File Inode Change Date/Time     : 2019:12:27 19:23:4
File Permissions                : rw-r--r--
File Type                       : JPEG
File Type Extension             : jpg
MIME Type                       : image/jpeg
JFIF Version                    : 1.01
Resolution Unit                 : inches
X Resolution                    : 300
Y Resolution                    : 300
Comment                         : ORZHS2BUMNVW2MYK
Image Width                     : 426
```

#1 Find the flag.

- Tried ORZHS2BUMNVW2MYK
- Realized it was encoded
- Used cyberchef



#1 Find the flag.

```
root@kali:~/Downloads# steghide extract -sf hide.jpg
Enter passphrase:

wrote extracted data to "flag-1.txt".

root@kali:~/Downloads# cat flag-1.txt
```

Steghide is a great tool to find some hidden data that couldn't be extracted using binwalk.

Note: steghide doesn't need password always

tryhackme{st3gh1d3_i5_l0v3}

[Part 4] #1 Find the flag.

#1 Find the flag.

Download: stegano.png

Hint:

Hiding data in LSB are a very common process. Especially in CTFs.

The most famous tool used for this is KDE68

P.S: Name of the tool is encrypted in a version of ROT cipher.

P.P.S: I repeat decode KDE68 to find the name of the tool.

(Hint look up ROT13 variants)

#1 Find the flag.

Decode KDE68

- Tried a bunch of different things until something worked

#1 Find the flag.

➔ You have to download zsteg

https://github.com/zed-0xff/zsteg

➔ Extract and run:
root@kali:~/zsteg-master# gem install zsteg

#1 Find the flag.

```
root@kali:~/Downloads# zsteg stegano.png
imagedata           .. text: "ywx46+%)"
b1,bgr,lsb,xy       .. text:
"=flag=4wbWyHV1VA43QJtvWdw8pLCwkADDQ7ZdYkz39KsKaXUeLtPy9DShWSp\n
....
```

#1 Find the flag.

I love cyberchef

Last build: 7 days ago - v9 supports multiple inputs and a Node API ... Options About / Support

Recipe

Remove non-alphabet chars

From Base58

Alphabet
123456789ABCDEFGH ...

Remove non-alphabet chars

From Base32

Alphabet
A-Z2-7=

Remove non-alphabet chars

Input

length: 55
lines: 1

4wbWyHV1VA43QJtvWdw8pLCwkADDQ7ZdYkz39KsKaXUeLtPy9DShWSp

Output

start: 24 time: 10ms
end: 24 length: 24
length: 0 lines: 1

tryhackme{lsb_4r3_l1t!!}

[Part 5] #1 Since you've been working hard..I wanted to hand out the flag to you but my dumb friend messed the whole image. Fix the image to get the flag.

#1 Fix the image.

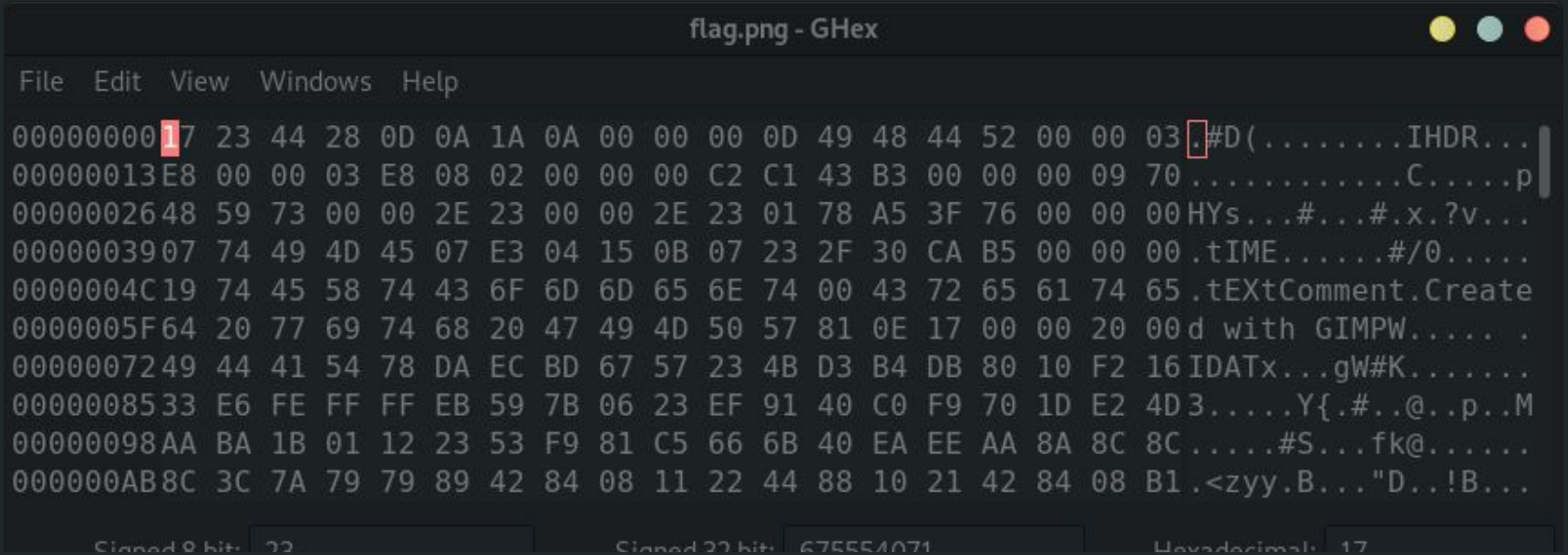
Download: flag.png

There are a lot of ways to mess a file. The most common one is to play with its headers.

NOTE: The flag is not in the tryhackme{}. For submission add tryhackme{} around the found flag.

#1 Fix the image.

Open the image with ghex and check the file signature: 17 23 44 28 0D 0A 1A ..



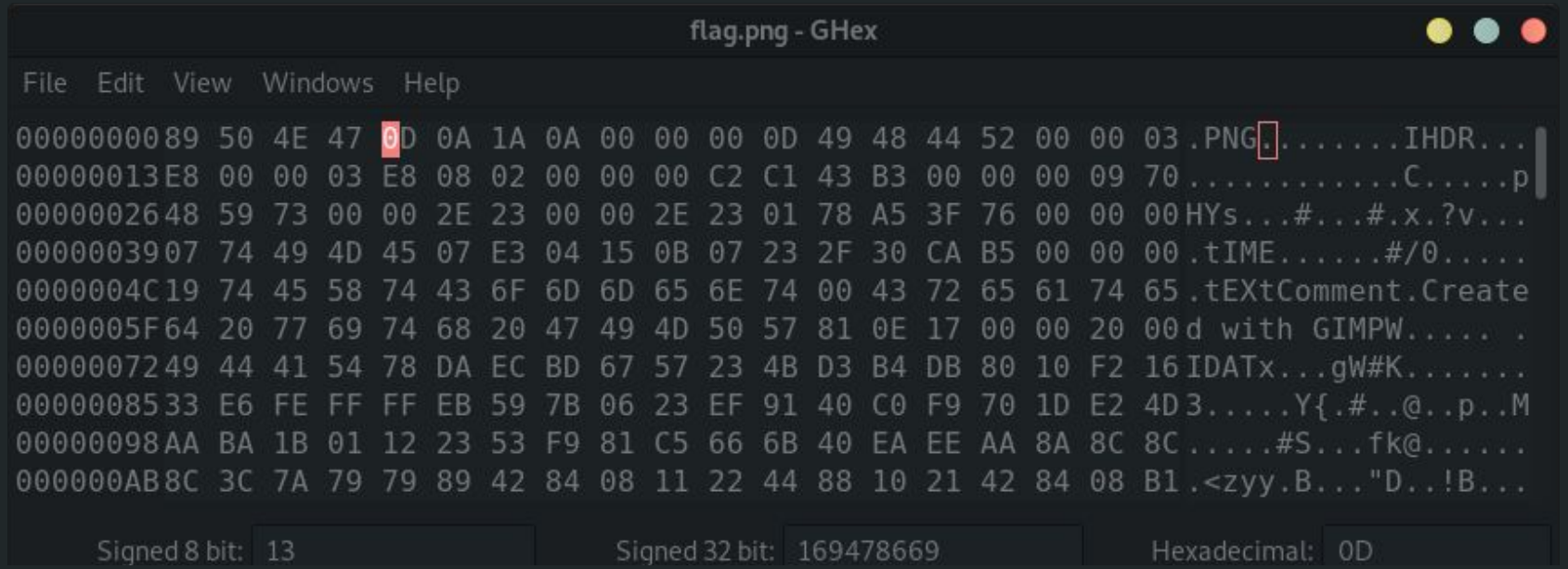
#1 Fix the image.

Lookup the file signature for png and compare it with:
17 23 44 28 0D 0A 1A ..

89 50 4E 47 0D 0A 1A 0A	.PNG....	0	png	Image encoded in the Portable Network Graphics format[13]

This doesn’t match.
Edit the hex on flag.png to match, and then save it.

#1 Fix the image.

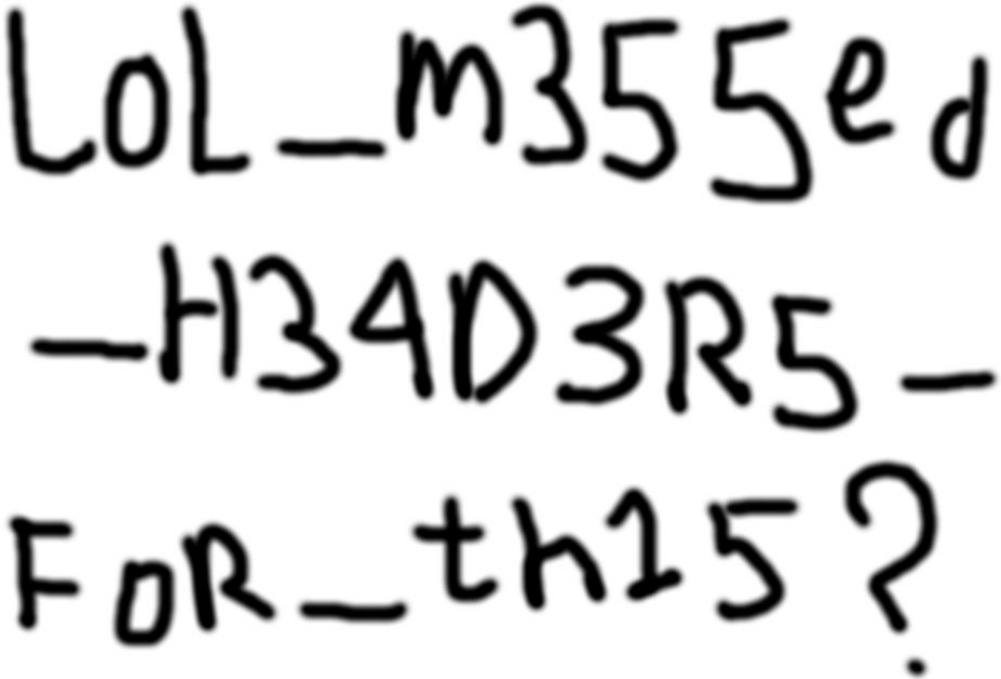


#1 Fix the image.

The fixed image is:

And the flag is:

tryhackme{LoL_m355ed_H34D3R5_FoR_th15?}



[Part 6] #1 Audio?!

#1 Audio?!

Download flag.wav
Hint:

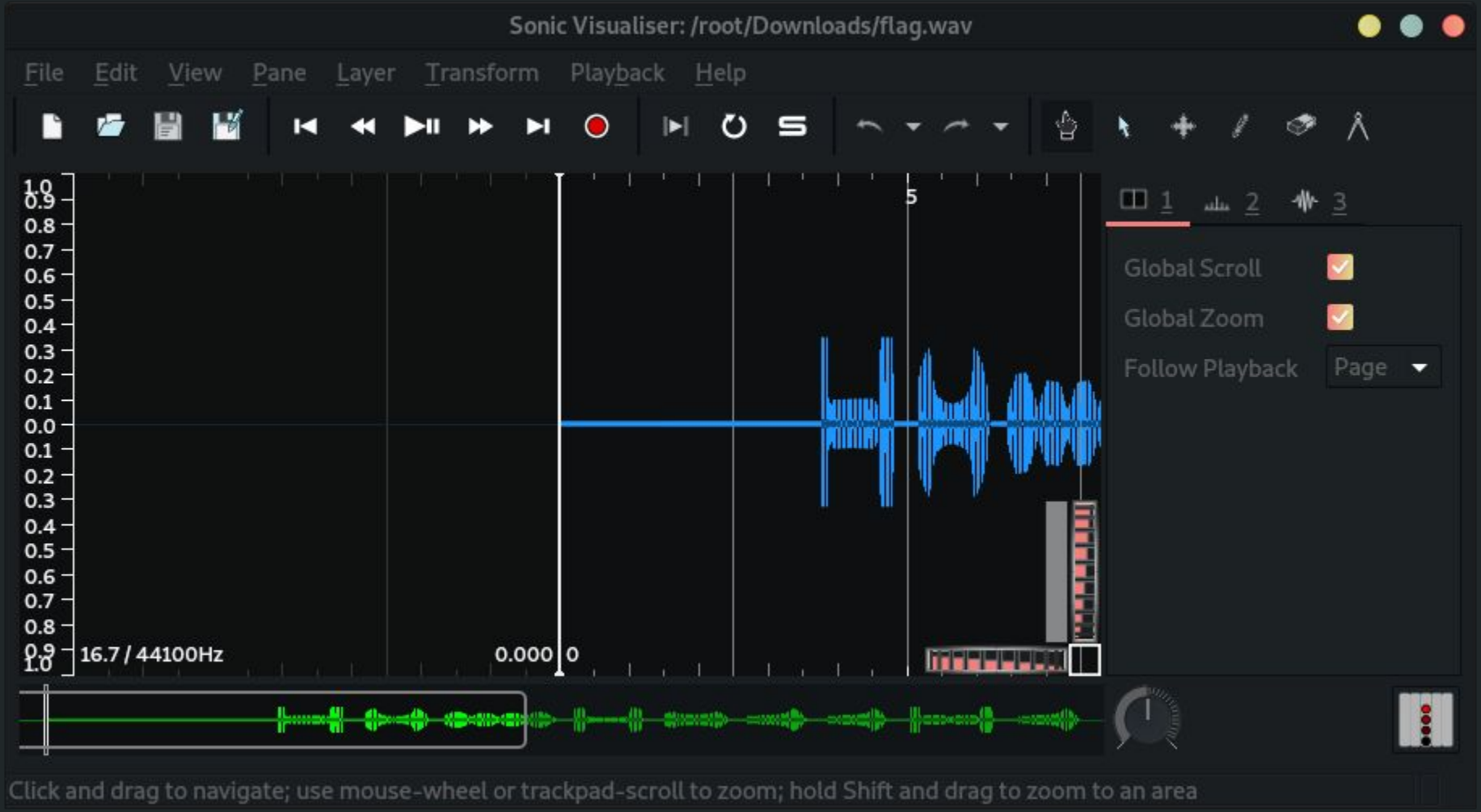
HACKER1: FBI is onto me that is why I am sending you a hidden message in an audio file.
HACKER2: What? Audio file...how the hell is that safe.
H1: It is because audio has nothing to do with it.
H2: So how can I see it.
H1: Just check the spectro.....
-----DISCONNECTED----------
This was the conversation intercepted by FBI between two hackers. FBI has provided you with the audio file can you help then find the message?

#1 Audio?!

In the hint it says check the spectro
After much searching I found a tool:

sonic-visualiser/kali-rolling 4.0-1 amd64
  viewing and analysing the contents of music audio files

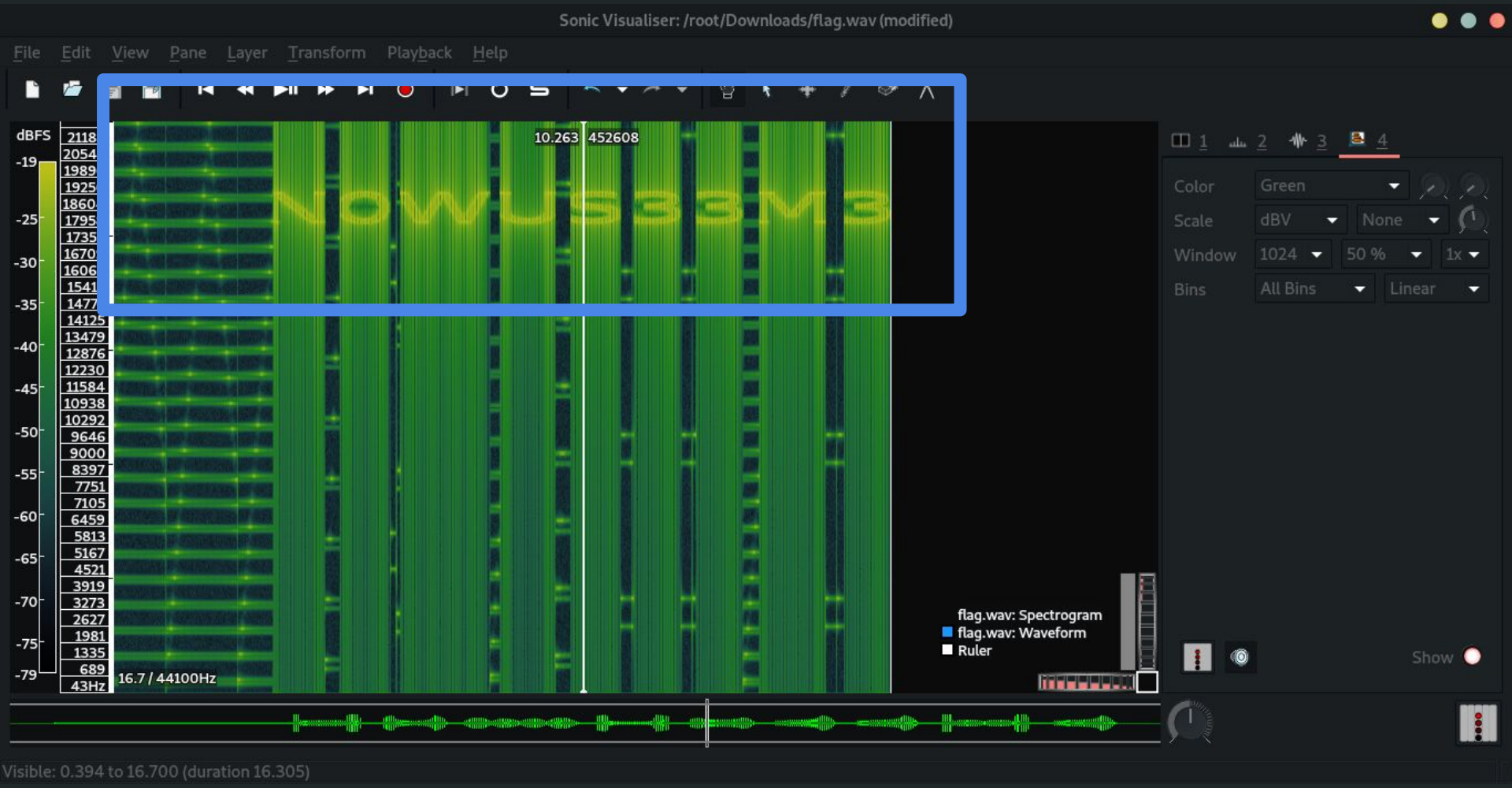
Downloaded it and opened the file


Sonic Visualiser: /root/Downloads/flag.wav
File
Edit
View
Pane
Layer
Transform
Playback
Help
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1.0
5
16.7 / 44100Hz
0.000 0
1
2
3
Global Scroll
Global Zoom
Follow Playback
Page
Click and drag to navigate; use mouse-wheel or trackpad-scroll to zoom; hold Shift and drag to zoom to an area

#1 Audio?!

To reveal the flag:

Layer > Add Spectrogram


Sonic Visualiser: /root/Downloads/flag.wav (modified)
File Edit View Pane Layer Transform Playback Help
10.263 452608
dBFS
16.7 / 44100Hz
flag.wav: Spectrogram
flag.wav: Waveform
Ruler
Color Green
Scale dBV None
Window 1024 50 % 1x
Bins All Bins Linear
Show
Visible: 0.394 to 16.700 (duration 16.305)

#1 Audio?!

That is so cool...

Flag is:

tryhackme{NOWUS33M3}

[Part 7] #1 Let's start with the basic

#1 Let's start with the basic

Let's start with the basic:

Aopz pz h Jhlzhy jpwoly zopmalk zlclu wvzpapvuz zv h pz lxbpchslua av o huk zv vu.

Doha fvb ullk pz h mshn ypnoa ayfohjrtl{Uv_jhlzhy_Uv_Jyfwav}

#1 Let's start with the basic

Text has been shifted. We have to figure out how much.

The last bit in the phrase is obviously the flag:

ayfohjrtl{Uv_jhlzhy_Uv_Jyfwav}

ayfohjrtl == tryhackme

#1 Let's start with the basic

Using ROT13

ROT13 - CyberChef - Mozilla Firefox

Home | My Drive - Google Drive | CCSC: Stego - Google S | image walk kali - Google | TryHackMe | functf | ROT13 - CyberChef

https://gchq.github.io/CyberChef/#recipe=ROT13(true,true,19)&input=YXlmb2hqcnRs

Kali Linux | Kali Training | Kali Tools | Kali Docs | Kali Forums | NetHunter | Offensive Security | Exploit-DB | GHDB | MSFU

Download CyberChef | Last build: 7 days ago - v9 supports multiple inputs and a Node API allowing you to program with Cyb… | Options | About / Support

Triple DES Encrypt
Triple DES Decrypt
RC2 Encrypt
RC2 Decrypt
RC4
RC4 Drop
ROT13
ROT47

Recipe

ROT13

- [x] Rotate lower case chars
- [x] Rotate upper case chars

Amount: 19

Input

length: 9
lines: 1

```
ayfohjrtl
```

Output

time: 3ms
length: 9
lines: 1

#1 Let's start with the basic

ROT13 - CyberChef - Mozilla Firefox

Home | My Drive - Google Drive | CCSC: Stego - Google S | image walk kali - Google | TryHackMe | functf | ROT13 - CyberChef

https://gchq.github.io/CyberChef/#recipe=ROT13(true,true,19)&input=YXlmb2hqcnRsCgpBb3

Kali Linux | Kali Training | Kali Tools | Kali Docs | Kali Forums | NetHunter | Offensive Security | Exploit-DB | GHDB | MSFU

Download CyberChef | Last build: 7 days ago - v9 supports multiple inputs and a Node API allowing you to program with Cyb… | Options | About / Support

Triple DES Encrypt
Triple DES Decrypt
RC2 Encrypt
RC2 Decrypt
RC4
RC4 Drop
ROT13
ROT47
XOR
XOR Brute Force

Recipe

ROT13

- [x] Rotate lower case chars
- [x] Rotate upper case chars

Amount: 19

Input

length: 155
lines: 5

```
ayfohjrtl

Aopz pz h Jhlzhy jpwoly zopmalk zlclu wvzpapvuz zv h pz lxbpchslua av o huk zv vu.
Doha fvb ullk pz h mshn ypnoa ayfohjrtl{Uv_jhlzhy_Uv_Jyfwav}
```

Output

time: 4ms
length: 155
lines: 5

```
tryhackme

This is a Caesar cipher shifted seven positions so a is equivalent to h and so on.
What you need is a flag right tryhackme{No_caesar_No_Crypto}
```

[Part 7] #2 Let's start with the basic

#2 Let's start with the basic

Guvf gvzr gurl ner fuvsgrq guvegrra cbfvgvbaf gung vf jul vg'f pnyyrq EBG guvegrra.

SYNT: gelunpxzr{ebg_guvegrra_vf_nyfb_pnrfne_pvcure}

#2 Let's start with the basic

Focusing on this

SYNT: gelunpxzr{ebg_guvegrra_vf_nyfb_pnrfne_pvcure}

SYNT is hint maybe?

➔ (Its actually flag)

➔ Using the same technique

#1 Let's start with the basic

ROT13 - CyberChef - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Home | My Drive - Google Drive | CCSC: Stego - Google S | image walk kali - Google | TryHackMe | functf | ROT13 - CyberChef

https://gchq.github.io/CyberChef/#recipe=ROT13(true,true,39)&input=U1lOVDogZ2VsdW5w

Kali Linux Kali Training Kali Tools Kali Docs Kali Forums NetHunter Offensive Security Exploit-DB GHDB MSFU

Download CyberChef

Last build: 7 days ago - v9 supports multiple inputs and a Node API allowing you to program with Cyb...

Options About / Support

Triple DES Encrypt
Triple DES Decrypt
RC2 Encrypt
RC2 Decrypt
RC4
RC4 Drop
ROT13
ROT47
XOR
XOR Brute Force

Recipe

ROT13

- [x] Rotate lower case chars
- [x] Rotate upper case chars

Amount: 39

Input

start: 154 end: 154 length: 0 | length: 154 lines: 6

```
SYNT: gelunpxzr
Guvf gvzr gurl ner fuvsgrq guvegrra cbfvgvbaf gung vf jul vg'f pnyyrq EBG guvegrra.
SYNT: gelunpxzr{ebg_guvegrra_vf_nyfb_pnrfne_pvcure}
```

Output

start: 154 end: 154 length: 0 | time: 17ms length: 154 lines: 6

```
FLAG: tryhackme
This time they are shifted thirteen positions that is why it's called ROT thirteen.
FLAG: tryhackme{rot_thirteen_is_also_caesar_cipher}
```

[Part 7] #3 What the hell is this?

#3 What the hell is this?

(@29]]]H:== E9:D 6G6C DE@An x >62? H6 42? ;FDE D9:7E E@ 2?J 2>@F?E @7 A@D:E:@?D H:E9 H926G6C 492C24E6C D6E]

ECJ924<>6Lu=2v0xD0p==0x0Hp?E0?@0q$N

#3 What the hell is this?

(@29]]]H:== E9:D 6G6C DE@An x >62? H6 42? ;FDE D9:7E E@ 2?J 2>@F?E @7 A@D:E:@?D H:E9 H926G6C 492C24E6C D6E]
ECJ924<>6Lu=2v0xD0p==0x0Hp?E0?@0q$N

➔ They have probably shifted more than just letters

⇒ The encoding that does that is ROT47

#3 What the hell is this?

ROT47 - CyberChef - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Home | My Drive - Google Drive | CCSC: Stego - Google S | image walk kali - Google | TryHackMe | functf | ROT47 - CyberChef

https://gchq.github.io/CyberChef/#recipe=ROT47(47)&input=KEAyOV1dXUg6PT0gRTk6RCA2

Kali Linux | Kali Training | Kali Tools | Kali Docs | Kali Forums | NetHunter | Offensive Security | Exploit-DB | GHDB | MSFU

Download CyberChef | Last build: 7 days ago - v9 supports multiple inputs and a Node API allowing you to program with Cyb... | Options | About / Support

Triple DES Encrypt
Triple DES Decrypt
RC2 Encrypt
RC2 Decrypt
RC4
RC4 Drop
ROT13
ROT47
XOR
XOR Brute Force

Recipe

ROT47

Amount
47

Input

start: 143 end: 143 length: 0 | length: 143 lines: 1

```
(@29]]]H:== E9:D 6G6C DE@An x >62? H6 42? ;FDE D9:7E E@ 2?J 2>@F?E @7 A@D:E:@?D H:E9
H926G6C 492C24E6C D6E] ECJ924<>6Lu=2v0xD0p==0x0Hp?E0?@0q$N
```

Output

start: 143 end: 143 length: 0 | time: 5ms length: 143 lines: 1

```
Woah...will this ever stop? I mean we can just shift to any amount of positions with
whaever character set. tryhackme{FlaG_Is_All_I_wAnt_no_BS}
```

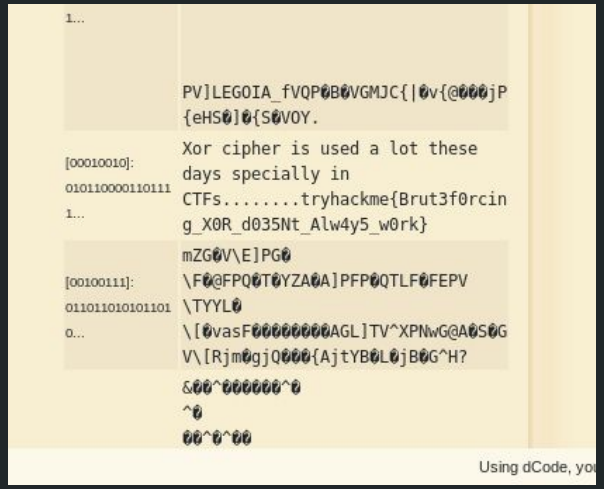
[Part 10] #1 Exclusive Or Random

#1 Exclusive Or Random

Hint: You know you can take any two beautiful messages(strings) and mesh them together and they'll come out complete random.

4a7d6032717b627a7760327b6132676177763273327e7d6632667a7761773276
736b6132616277717b737e7e6b327b7c32514654613c3c3c3c3c3c3c66606b7
a7371797f7769506067662174226071 7b7c754d4a22404d762221275c664d537
e65266b274d652260796f18

#1 Exclusive Or Random



You have to brute force it:

[00010010]: 01011000011011111...Xor cipher is used a lot these days specially in CTFs.....tryhackme{Brut3f0rcing_X0R_d035Nt_Alw4y5_w0rk}

[Part 8] #1 Ancient Times

#1 Ancient Times

#1 Ancient Times

- Looked around until I found out what it was
- Pigpen cipher
- Found a site to decode it

Kali Linux Kali Training Kali Tools Kali Docs Kali Forums NetHunter Offensive Security

Search for a tool

SEARCH A TOOL ON DCODE BY KEYWORDS:

e.g. type random GO

Results

↑↓	↑↓
(Original) #,#•,×,×•	LOOKEDLIKESOMEALIENLANGUAGETOMETRYHACKMEPIGANDPEN
#,×,#•,×•	PSSOEDPIOEJSQEAPIERPARGLAGEKSQEKVYHACOQETIGARDTER
La Buse	EKKCJHERCJTKGJBERJIEBINVBNJTKGJTQUPBFCGJMRNBIHMJI
Heinrich von Nettelsheim	JMMKEDJIKEPMKEAJIELJALGUAGEQMKEQOYHACKKENIGALDNEL
#•,×•,#,×	CFFBRQCVBRWFDRNCVRECNETYNTRXFDRXILUNPBDRGVTNEQGRE
	TWWSMLTQSMAWUMITQMVTIVOCIOMBWUMBZ

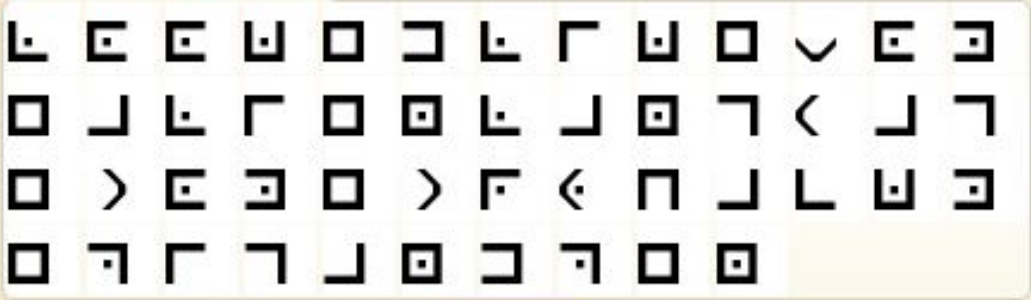
Pigpen Cipher

Sponsored ads

PigPen Decoder

SYMBOLS OF THE PIGPEN ALPHABET (CLICK TO ADD)

PIG-PEN CIPHERTEXT



DECRYPT

FORENSICS 02 : FIND THE FLAG

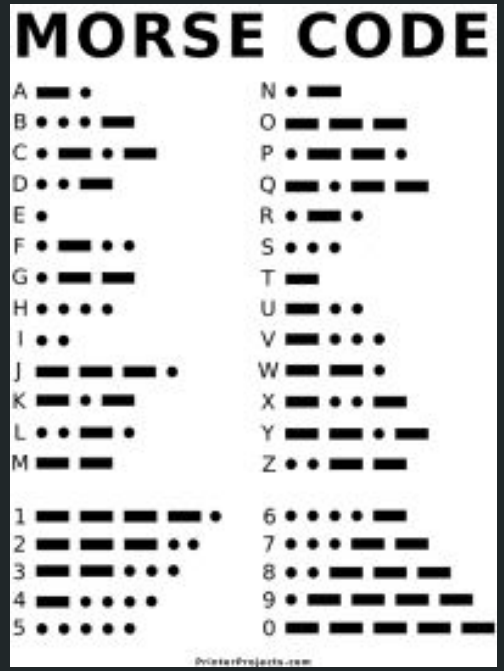
FORENSICS 02: THE FLAG

Check out f.02.wav

```
crazyeights@kali: ~/Downloads/ctf_primer_01/forensics
crazyeights@kali:~/Downloads/ctf_primer_01/forensics$ exiftool -a f.02.wav
ExifTool Version Number         : 11.80
File Name                       : f.02.wav
Directory                       : .
File Size                       : 394 kB
File Modification Date/Time     : 2019:10:26 13:57:46-04:00
File Access Date/Time           : 2020:02:02 19:41:43-05:00
File Inode Change Date/Time     : 2020:02:02 19:41:41-05:00
File Permissions                : rw-rw-r--
File Type                       : WAV
File Type Extension             : wav
MIME Type                       : audio/x-wav
Encoding                        : Microsoft PCM
Num Channels                    : 1
Sample Rate                     : 11050
Avg Bytes Per Sec               : 11050
Bits Per Sample                 : 8
Duration                        : 0:00:37
crazyeights@kali:~/Downloads/ctf_primer_01/forensics$
```

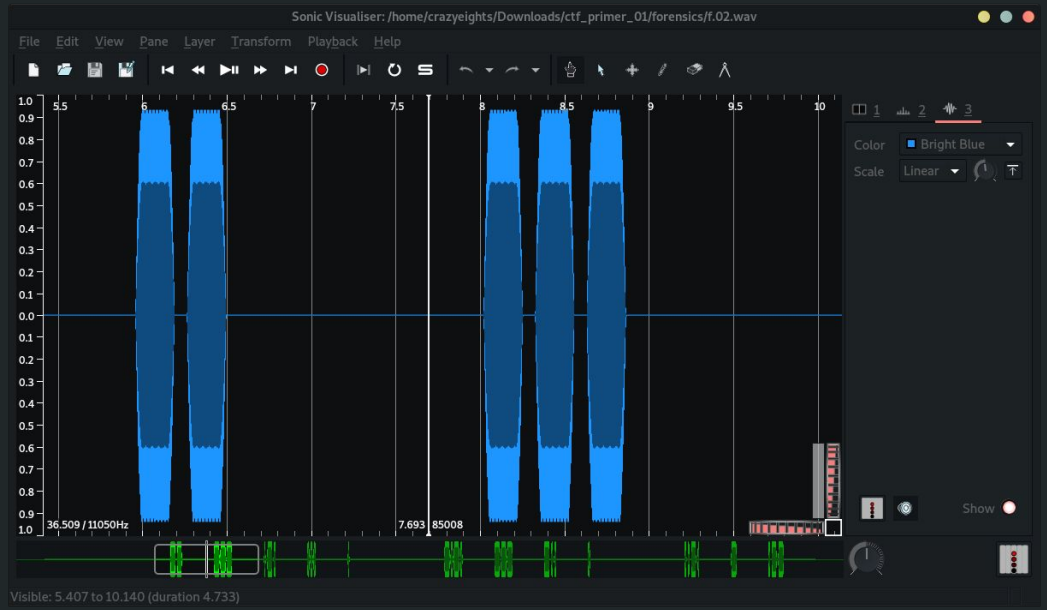
FORENSICS 02: THE FLAG

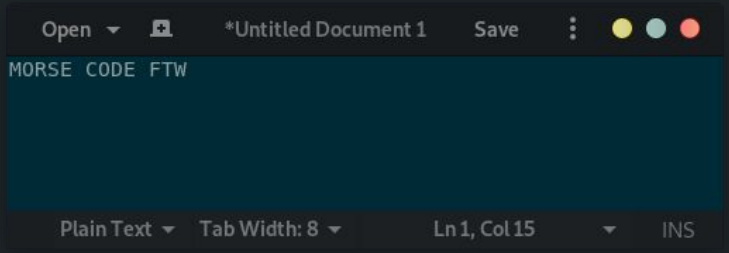
Listening to it you hear very loud beeps. Hmmm could it be morse?



FORENSICS 02: THE FLAG

Opening it with Sonic Visualiser, and translating from morse based on size of sound wave:





Forensics 10: Could it be the brain??

Forensics 10: Could it be the brain?

Open

f.10

~/Downloads/ctf_primer_01/forensics

Save

```
-[------->+<]>---.++++++.-----------.++++++.>--[-->+++++<]>.[->++++++<]>--.[--->+<]>----.----.+++++.+++[->+++<]>.+++++++++++
++.----------.------.----.++++++++++++.-----------.++++++++++++.--------------.++++++.--[--->+<]>--.[->+++<]>++.++++++.--.
[--->+<]>----.++[->+++<]>.++.--[--->+<]>---.--------------.------.++++.++++++++++++..----.[----->+<]>+.
```

Forensics 10: Could it be the brain?

```
1. ++++.--[--->+<]>--.[->+++<]>++.++++++.--.[--->+<]>----.++[->+++<]>.++.--[--->+<]>---.-------------.------.++++.++++++++++++..---.[----->+<]>+.
```

About

A Brainfuck editor & optimizing interpreter, written in JavaScript. The code is converted to JavaScript code and then run in a web worker, which speeds up execution at lot (try mandelbrot.b for example).

Load example program

Have fun! :-)

You can add special chars to the input field:
Decimal: **\65** (same as "A")
Hexadecimal: **\x7E** (same as "~")
Control characters: **\r \n \t**

For any kind of feedback, toss me a mail to

```
--[----->+<]>---.++++++++++++.+.++++++++.+[-->+<]>+++.++
[-->+++<]>.++++++++++++.+.++++++++.-[-->++++++<]>++.[--->++
<]>-.------------.
```

Links:

- Wikipedia on Brainfuck
- The Brainfuck archive
- Brainfuck snippets
- Text generator
- BF code compressor

run | stop | load from server | link to this code | view memory | view generated code | minify

Finished in 9 ms.

```
FLAG{esoteric_lanaguages_are_cool}
```

Options

Cell size (Bits): 8 16 32

Dynamic (infinite) Memory:

Memory size: 30000

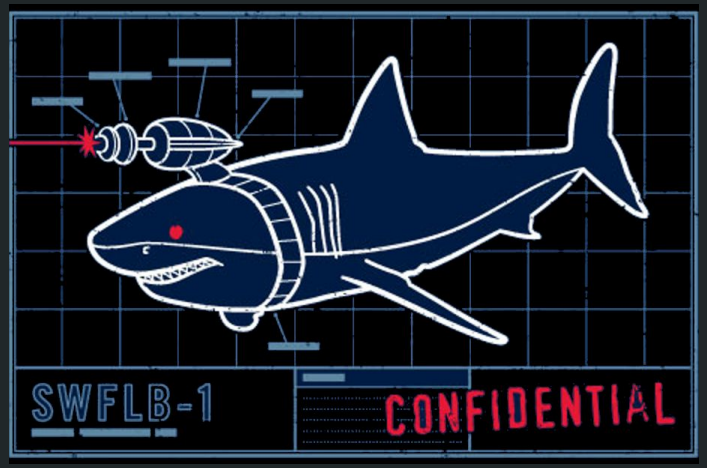
Memory overflow behaviour:

undefined (fast) wrap abort

[Part 12]: What is in the bin?

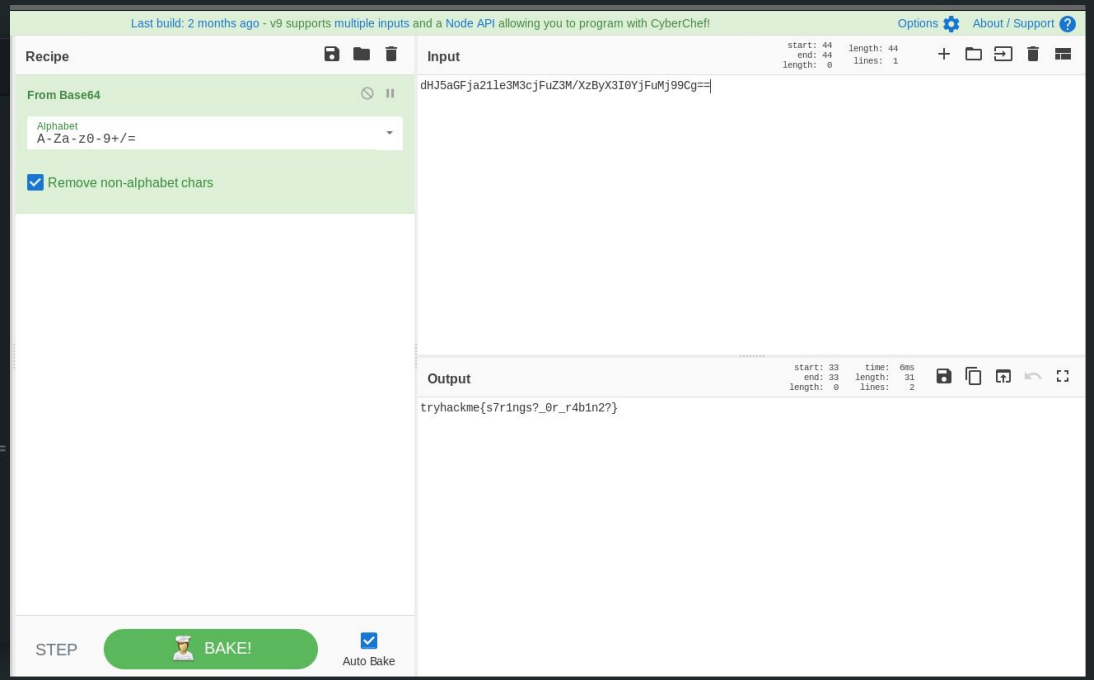
[Part 12] What is in the bin?

- Download flag binary file
- Find the hidden flag



[Part 12] What is in the bin?

```
crazyeights@kali: ~/Downloads
crazyeights@kali:~$ cd Downloads/
crazyeights@kali:~/Downloads$ strings flag
/lib64/ld-linux-x86-64.so.2
libc.so.6
puts
printf
__cxa_finalize
__libc_start_main
GLIBC_2.2.5
_ITM_deregisterTMCloneTable
__gmon_start__
_ITM_registerTMCloneTable
u3UH
[]A\A]A^A_
Oh so you are here to get the flag huh?!
bG9sLi55b3UgdGhvdWdodCB0aGlzIHdvdWxkIGJlIHRoYXQgZWFzeSBodWg/Cg==
Well Good luck!
dHJ5aGFja21le3M3cjFuZ3M/XzByX3I0YjFuMj99Cg==
;*3$"
GCC: (GNU) 8.2.1 20180831
GCC: (GNU) 8.2.1 20181127
init.c
crtstuff.c
deregister_tm_clones
```



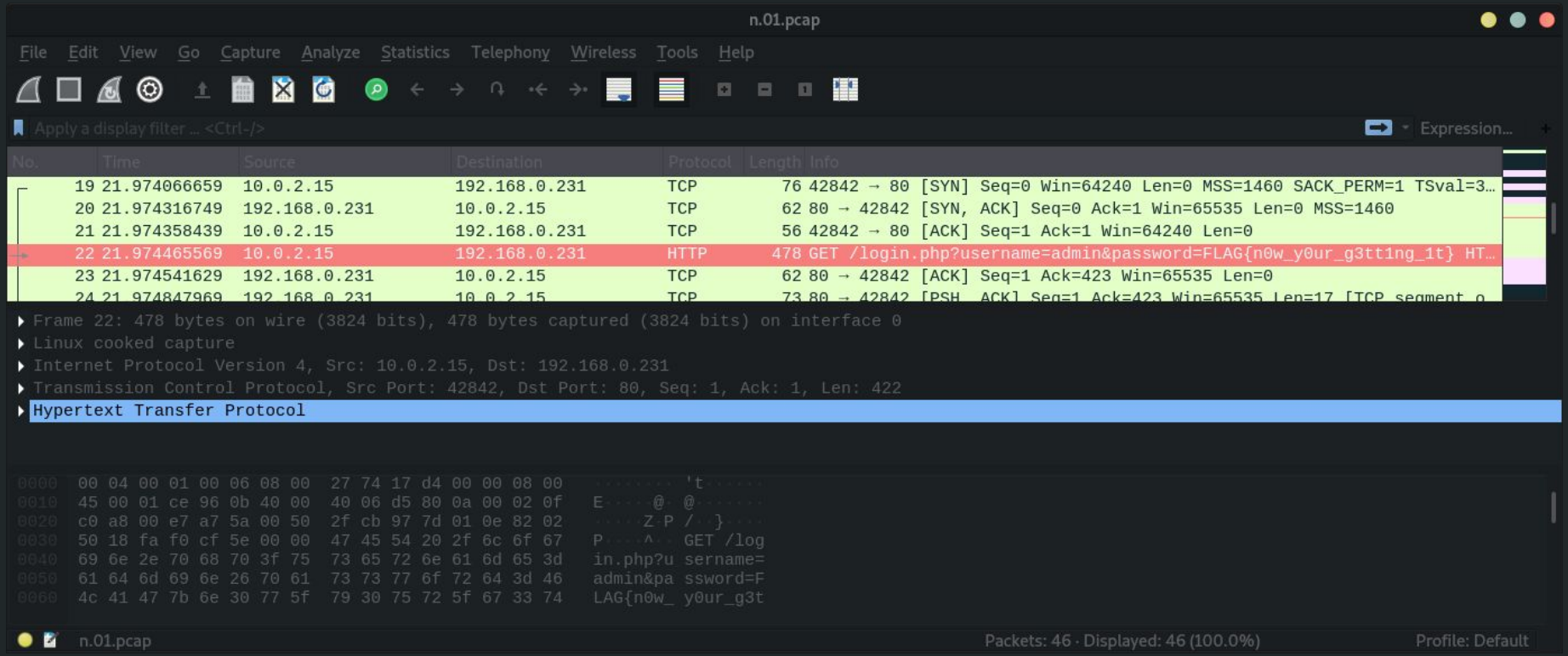
Analysing PCAPs Part 1

PCAP1

Download n.01.pcap

- Open with wireshark
- Hunt for flag in conversation

PCAP1


n.01.pcap
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...
No. Time Source Destination Protocol Length Info
19 21.974066659 10.0.2.15 192.168.0.231 TCP 76 42842 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=3…
20 21.974316749 192.168.0.231 10.0.2.15 TCP 62 80 → 42842 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1460
21 21.974358439 10.0.2.15 192.168.0.231 TCP 56 42842 → 80 [ACK] Seq=1 Ack=1 Win=64240 Len=0
22 21.974465569 10.0.2.15 192.168.0.231 HTTP 478 GET /login.php?username=admin&password=FLAG{n0w_y0ur_g3tt1ng_1t} HT…
23 21.974541629 192.168.0.231 10.0.2.15 TCP 62 80 → 42842 [ACK] Seq=1 Ack=423 Win=65535 Len=0
24 21.974847969 192.168.0.231 10.0.2.15 TCP 73 80 → 42842 [PSH, ACK] Seq=1 Ack=423 Win=65535 Len=17 [TCP segment o
Frame 22: 478 bytes on wire (3824 bits), 478 bytes captured (3824 bits) on interface 0
Linux cooked capture
Internet Protocol Version 4, Src: 10.0.2.15, Dst: 192.168.0.231
Transmission Control Protocol, Src Port: 42842, Dst Port: 80, Seq: 1, Ack: 1, Len: 422
Hypertext Transfer Protocol
0000 00 04 00 01 00 06 08 00 27 74 17 d4 00 00 08 00 ..... 't.....
0010 45 00 01 ce 96 0b 40 00 40 06 d5 80 0a 00 02 0f E.....@. @.....
0020 c0 a8 00 e7 a7 5a 00 50 2f cb 97 7d 01 0e 82 02 .....Z.P /..}....
0030 50 18 fa f0 cf 5e 00 00 47 45 54 20 2f 6c 6f 67 P....^.. GET /log
0040 69 6e 2e 70 68 70 3f 75 73 65 72 6e 61 6d 65 3d in.php?u sername=
0050 61 64 6d 69 6e 26 70 61 73 73 77 6f 72 64 3d 46 admin&pa ssword=F
0060 4c 41 47 7b 6e 30 77 5f 79 30 75 72 5f 67 33 74 LAG{n0w_ y0ur_g3t
n.01.pcap
Packets: 46 · Displayed: 46 (100.0%)
Profile: Default

Analysing PCAPs Part 2

PCAP2

Download n.02.pcap

- Open with wireshark
- Hunt for flag in conversation

PCAP2

Wireshark · Follow HTTP Stream (tcp.stream eq 3) · n.02.pcap

```
POST /login.php HTTP/1.1
Host: 192.168.0.231
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
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: keep-alive
Upgrade-Insecure-Requests: 1
Pragma: no-cache
Cache-Control: no-cache
Content-Length: 46

username=admin&password=FLAG{1_am_th3_p0stm4n}<head>
<title>Error response</title>
</head>
<body>
<h1>Error response</h1>
<p>Error code 501.
<p>Message: Unsupported method ('POST').
<p>Error code explanation: 501 = Server does not support this operation.
</body>
```

Analysing PCAPs Part 3

PCAP3

Download n.03.pcap

- Open with wireshark
- Hunt for flag in conversation

PCAP3

Wireshark · Follow TCP Stream (tcp.stream eq 3) · n.03.pcap

```
GET /help.php HTTP/1.1
Host: 192.168.0.231
User-Agent: Mozilla/5.0 (FLAG{s3cr3t_ag3nt}) Gecko/20100101 Firefox/60.0
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: keep-alive
Upgrade-Insecure-Requests: 1
Pragma: no-cache
Cache-Control: no-cache

HTTP/1.0 200 OK
Server: SimpleHTTP/0.6 Python/2.7.17rc1
Date: Sat, 26 Oct 2019 11:19:28 GMT
Content-type: application/octet-stream
Content-Length: 14
Last-Modified: Sat, 26 Oct 2019 11:19:22 GMT

Help yourself
```

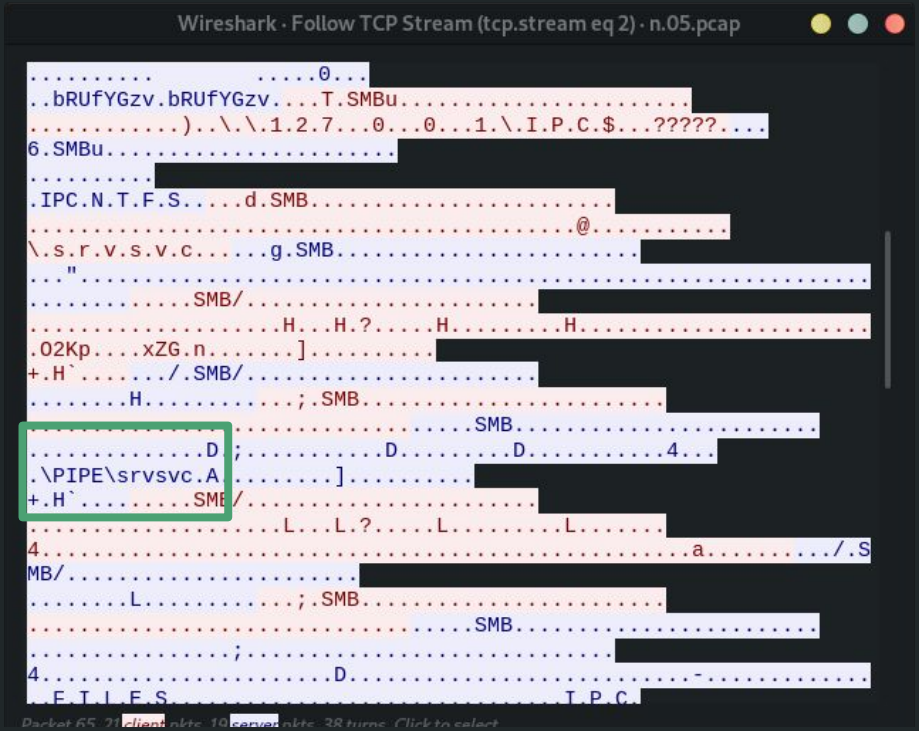
Analysing PCAPs Part 4

PCAP4

Download n.05.pcap

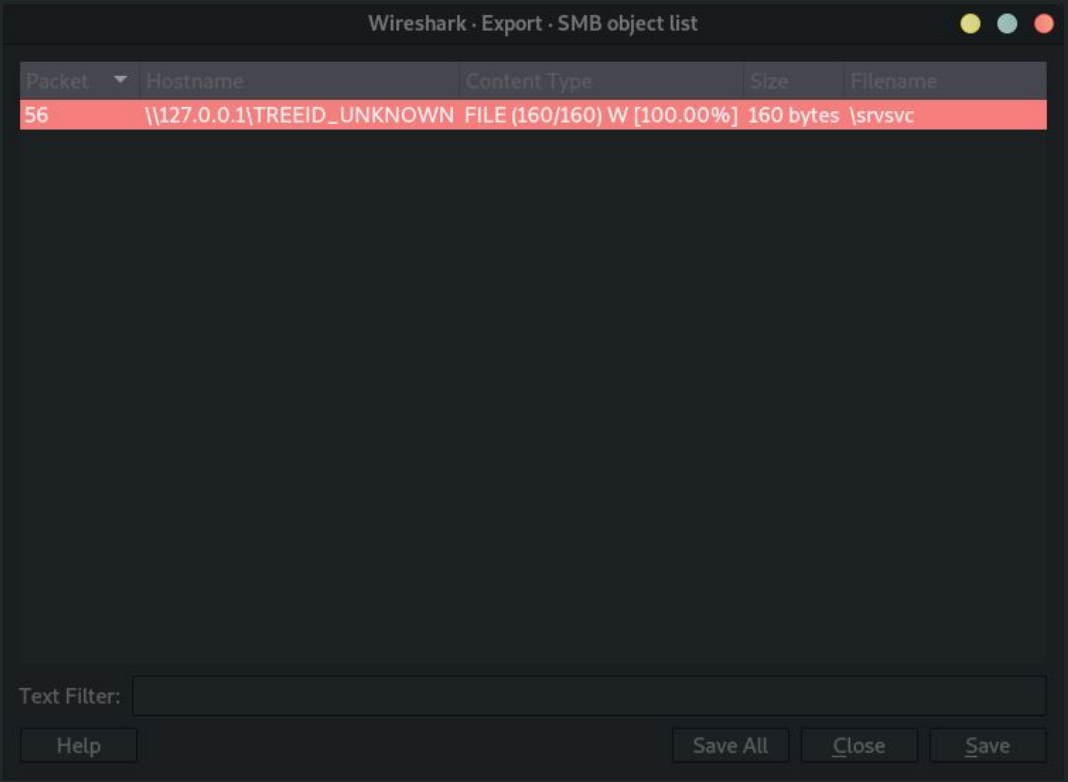
- Open with wireshark
- Find and export the file from the capture

PCAP4


Wireshark · Follow TCP Stream (tcp.stream eq 2) · n.05.pcap
.....0...
..bRUfYGzv.bRUfYGzv....T.SMBu.....
.....)..\.\.1.2.7...0...0...1.\.I.P.C.$...?????....
6.SMBu.....
.....
.IPC.N.T.F.S.....d.SMB.....
.....@.....
\.s.r.v.s.v.c.....g.SMB.....
..."
.....SMB/.....
.....H...H.?.....H.....H.....
.O2Kp....xZG.n.....].....
+.H`...../.SMB/.....
.....H.....;.SMB.....
.....SMB.....
.....D;.....D.....D.....4...
.\PIPE\srvsvc.A.....].....
+.H`.....SMB/.....
.....L...L.?.....L.....L.....
4.....a...../.S
MB/.....
.....L.....;.SMB.....
.....SMB.....
.....;.....
4.....D.....-.....
..F.I.L.E.S.....I.P.C.

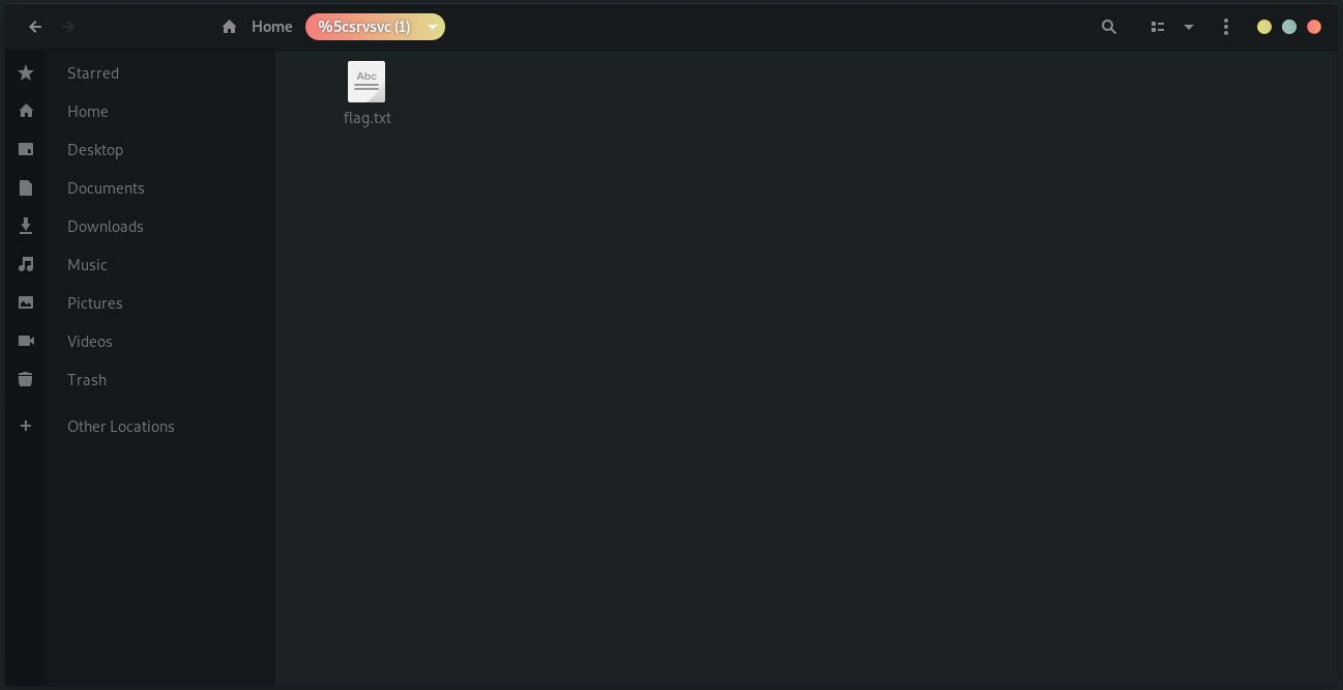
PCAP4

Go to File > Export Objects > SMB



PCPAP4

Decompress the archive to get the flag



BONUS 1: [Part 7] #4 What the hell is this?

#4 What the hell is this?

Fmeorcbi gc rmd gyowyb sp sw gd. Afy gybiq gi hewr geld xfo jjkk rbcfkgiwi{TsKcxipo_gGzLcB_mQ_MeCcep_mmNrlp}

P.S: Don't forget to use your brain ;)

#4 What the hell is this?

Keeping only the flag part:

Three different amounts that were shifted by:

tdehmikyk{VuMezkrq_iIbNeD_oS_OgEegr_ooPtKr}

hrsvawymy{JiAsnyfe_wWpBsR_cG_CuSsuf_ccDhYf}

nxybgcese{PoGytelk_cCvHyX_iM_IaYyal_iiJnEl}

--> You can see that together they make tryhackme

#4 What the hell is this?

--> Replace the characters that are in the wrong position with # to make clearer

t##h##k##{V##e##r#_#l##e#r_#s_##E#r_##P##r}

#r##a##m#{#i##n##e_##p##R_##_C##s##_c##h##}

##y##c##e{##G##e##_c##H##_i#_#a##a#_#i##E#}

the flag is:

tryhackme{ViGenere_cIpHeR_iS_CaEsar_ciPhEr}

--> I figured this out with pencil and paper I am sure there is faster way

BONUS 3: [Part 9]: #1 Genetics

#1 Genetics

I heard scientist found ways to hide data in DNA and stuff. Is it really true?

CTCAAAATAATCTTGATTACAATGATTAGTACATTGAAACACACATTGCCACAG
AGAATCACGTTGAAAATCCGACATACTAGAATCACGTTGCATATGTTGATAAAA
AGGACATTGCATACTACAAGACACTTGATAACACAGCAGAAAACGACATTGCA
GACAAAGCCACACACATTTTTGTAAACAAGTAGTTTGCCGACTATGTTGAAGAA
ACACACACAGTTGGAGTTGAGCCCACAGCATTTGATCACAACAAATTTGATAC
GATTGAATAAAATAATCTTGACCAGTAAAACGTTGGAAACAGCTTCGCATTCAA
AGTGAGACCCAGAC

#1 Genetics

(This actually took me a long time to figure out)

First I looked up dna code to english and scrolled until I found this promising table:

DNA CODE

Codon	English	Codon	English	Codon	English	Codon	English
AAA	a	CAA	q	GAA	G	TAA	W
AAC	b	CAC	r	GAC	H	TAC	X
AAG	c	CAG	s	GAG	I	TAG	Y
AAT	d	CAT	t	GAT	J	TAT	Z
ACA	e	CCA	u	GCA	K	TCA	1
ACC	f	CCC	v	GCC	L	TCC	2
ACG	g	CCG	w	GCG	M	TCG	3
ACT	h	CCT	x	GCT	N	TCT	4
AGA	i	CGA	y	GGA	O	TGA	5
AGC	j	CGC	z	GGC	P	TGC	6
AGG	k	CGG	A	GGG	Q	TGG	7
AGT	l	CGT	B	GGT	R	TGT	8
ATA	m	CTA	C	GTA	S	TTA	9
ATC	n	CTC	D	GTC	T	TTC	0
ATG	o	CTG	E	GTG	U	TTG	space
ATT	p	CTT	F	GTT	V	TTT	. (period)

#1 Genetics

I started manually translating it, but then gave up and wrote a program to do it for me.

Open ▾ dna_code.py ~/

```
#DNA TRIPLES
#Input: DNA triples without spaces
dnac={'AAA': 'a',
      'AAC': 'b',
      'AAG': 'c',
      'AAT': 'd',
      'ACA': 'e',
      'ACC': 'f',
      'ACG': 'g',
      'ACT': 'h',
      'AGA': 'i',
      'AGC': 'j',
      'AGG': 'k',
      'AGT': 'l',
      'ATA': 'm',
      'ATC': 'n',
      'ATG': 'o',
```

```
      'TTC': '0',
      'TTG': ' ',
      'TTT': '.'};

def get_input():
        code=raw_input("Enter dna triples:")
        return code


def dna_code_to_english(code):
        plain=''
        for i in range(0, len(code), 3):
                plain+=dnac[code[i:i+3]]
        return plain

code=get_input()
plain=dna_code_to_english(code)
print("Output:")
print(plain)
```

#1 Genetics

Program Output:

```
root@kali: ~
root@kali:~# python dna_code.py
Enter dna triples:CTCAAAATAATCTTGATTACAATGATTAGTACATTGAAACACACATTGCCACAGAGAATCAC
GTTGAAAATCCGACATACTAGAATCACGTTGCATATGTTGATAAAAAGGACATTGCATACTACAAGACACTTGATAACAC
AGCAGAAAACGACATTGCAGACAAAGCCACACACATTTTTGTAAACAAGTAGTTTGCCGACTATGTTGAAGAAACACACA
CAGTTGGAGTTGAGCCCACAGCATTTGATCACAACAAATTTGATACGATTGAATAAAATAATCTTGACCAGTAAAACGTT
GGAAACAGCTTCGCATTCAAAGTGAGACCCAGAC
Output:
Damn people are using anything to make their message secure. Well who cares I ju
st need my damn flag GeN3t1c5HuH
root@kali:~#
```

Flag: tryhackme{GeN3t1c5HuH}

BONUS 4: [Part 11] #1 Morse Code

#1 Morse Code

Download the file: morse.txt

Hint: Morse code is being used for a very long time. And since then there has been a lot of versions like using your eyebrows, flashing torches, tapping etc.

#1 Morse Code

Found a table that had the conversion:

MorseCode.World

International | American | More | SCPhillips.com

the highlighted letters or symbols the Morse sound will be played.

Letter	Morse
A	di-dah
B	dah-di-di-dit
C	dah-di-dah-dit
D	dah-di-dit
E	dit
F	di-di-dah-dit
G	dah-dah-dit
H	di-di-di-dit
I	di-dit

Letter	Morse
N	dah-dit
O	dah-dah-dah
P	di-dah-dah-dit
Q	dah-dah-di-dah
R	di-dah-dit
S	di-di-dit
T	dah
U	di-di-dah
V	di-di-di-dah

Digit	Morse
0	dah-dah-dah-dah-dah
1	di-dah-dah-dah-dah
2	di-di-dah-dah-dah
3	di-di-di-dah-dah
4	di-di-di-di-dah
5	di-di-di-di-dit
6	dah-di-di-di-dit
7	dah-dah-di-di-dit
8	dah-dah-dah-di-dit

Sound Controls

Pitch
550

Speed
20

Farnsworth speed
20

#1 Morse Code

Wrote a program to perform the decoding:

```
root@kali:~# python morse_to_en.py
Enter morse code:dah dah-dah-dah di-dah-di-dit dah-di-dit dah-di-dah-dah dah-dah
-dah di-di-dah dah di-di-di-dit dit dah-di-dah-dah di-dah di-dah-dit dit di-di-d
ah di-di-dit di-dit dah-dit dah-dah-dit di-dah dah-dit dah-di-dah-dah dah di-di-
di-dit di-dit dah-dit dah-dah-dit dah dah-dah-dah dit dah-dit dah-di-dah-dit di-
dah-dit dah-di-dah-dah di-dah-dah-dit dah dah di-di-di-dit dit di-di-dit dit dah
-di-dit di-dah dah-di-dah-dah di-di-dit di-dah-di-dah-di-dah di-di-dah-dit di-da
h-di-dit di-dah dah-dah-dit di-dit di-di-dit di-dit dah-dit dah di-di-di-dah-dah
 di-dah-dit dah-dit di-di-di-di-dah dah di-dit dah-dah-dah-dah-dah dah-dit di-di
-di-di-dah di-dah-di-dit dah-dah dah-dah-dah-dah-dah di-dah-dit di-di-dit di-di-
di-dah-dah dah-di-dah-dit dah-dah-dah-dah-dah dah-di-dit di-di-di-dah-dah
Output:
TOLDYOUTHEYAREUSINGANYTHINGTOENCRYPTTHESEDAYS.FLAGISINT3RN4TI0N4LM0RS3C0D3
root@kali:~#
```

```python
'dah-di-dah-di-dah-dah': '!',
'di-dah-di-dah-di-dah':'.',
'dah-di-di-di-di-dah':'-',
'di-dah-di-dah-dit':'+',
'di-dah-di-di-dah-dit':'"',
'di-di-dah-dah-di-dit':'?',
'dah-di-di-dah-dit':'\\'
};

def get_input():
        code=raw_input("Enter morse code:")
        return code


def morse_code_to_english(code):
        plain=''
        code_arr=code.split(' ')
        for i in range(0, len(code_arr)):
                plain+=morse_dict[code_arr[i]]
        return plain

code=get_input()
plain=morse_code_to_english(code)
print("Output:")
print(plain)
```

End.

There are about 5 more. But they are about reverse engineering, and are unrelated to the topic.

There are a ton of challenges similar to these on this site, and hack the box.

I used to do these challenges in first year:

https://cryptopals.com/sets/1/challenges/

https://www.mysterytwisterc3.org/en/challenges/