CTF HACKON 2021
STEGO – THEY ARE COMING FOR US

HackOn 2021

https://www.hackon.es/
@HackOnURJC

ENUNCIADO

My friend J.P. García has sent to me this image. He says that it has a hidden message, but the file seems to be corrupt and I can not recover it. Could you help me to get the message?

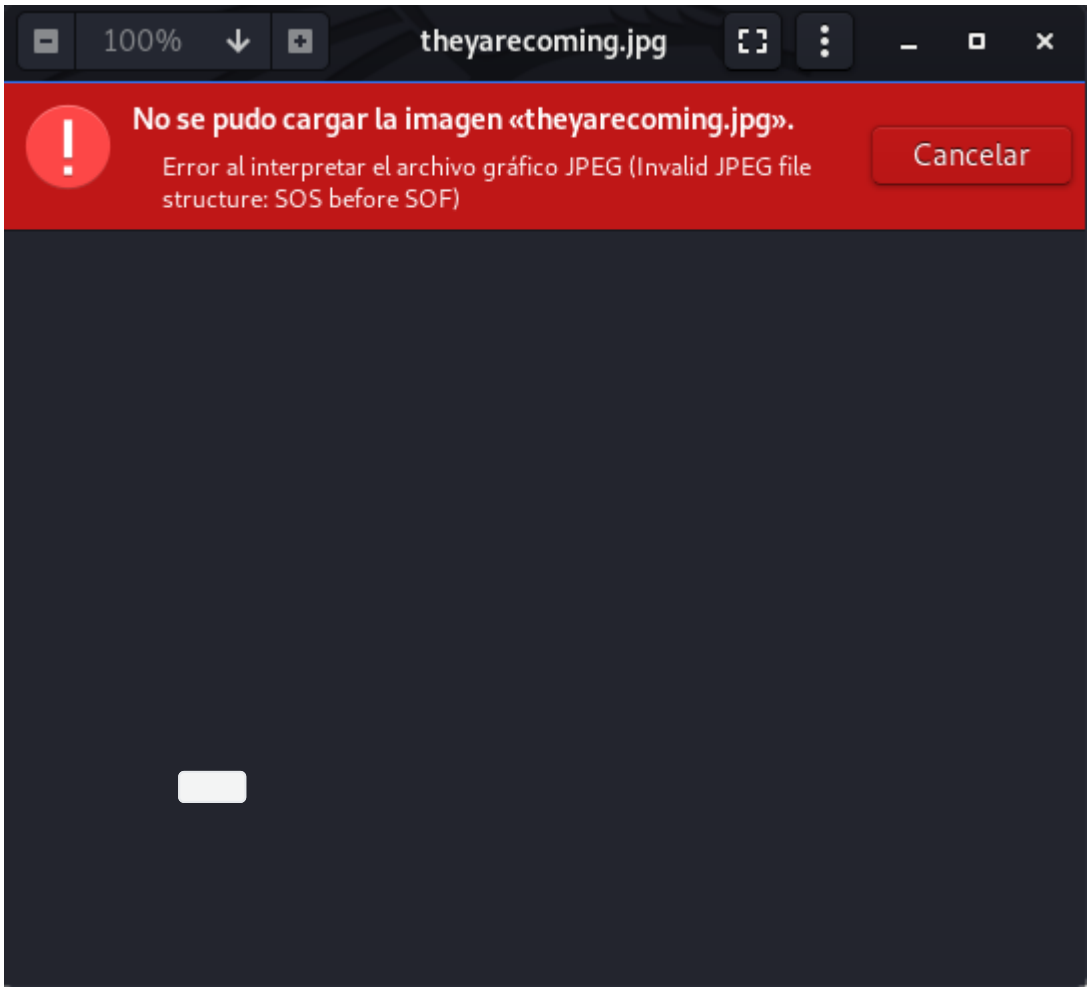
FLAG

HackOn{5p3ct0gr4m_4_Th3_W1n}

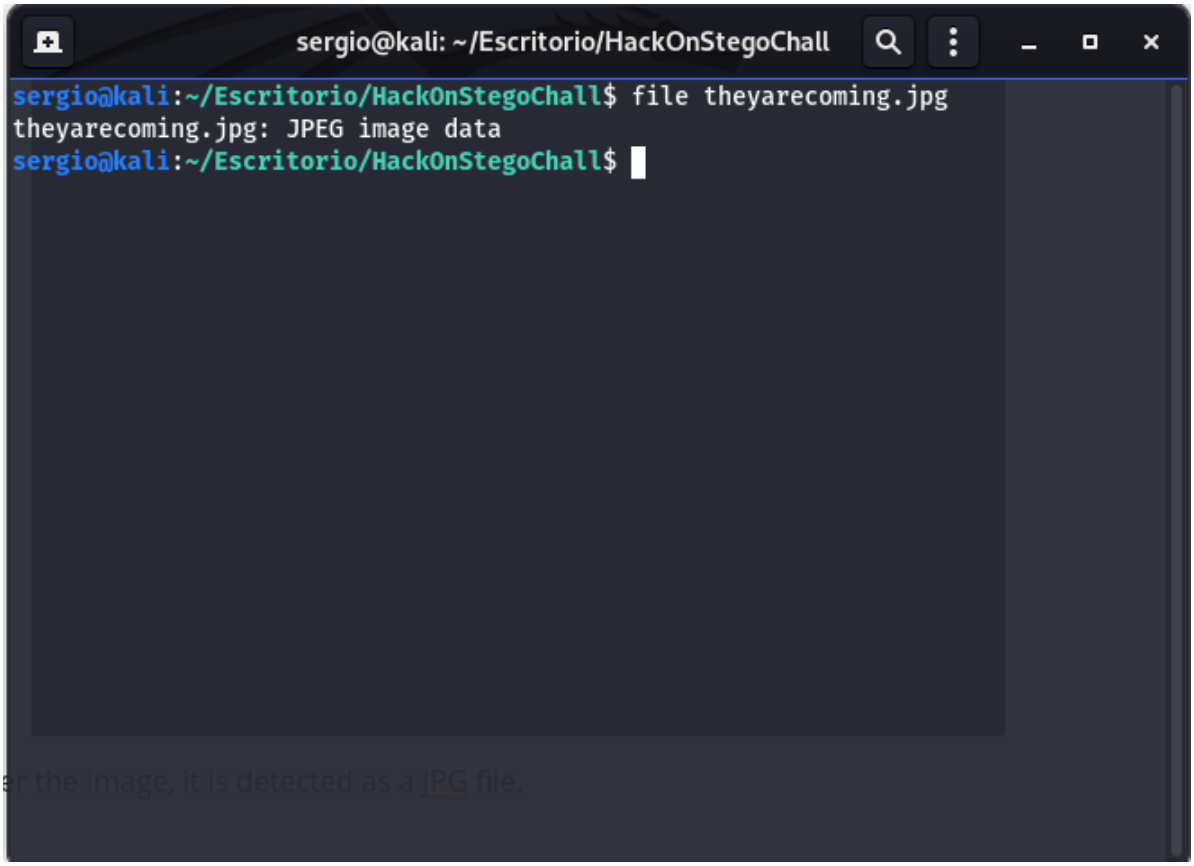
SOLUCIÓN

First steps

As we can see, if we download the image and try to open it, we have an error message that indicate us that the file is corrupted:



And executing the `file` command over the image, it is detected as a JPG file.



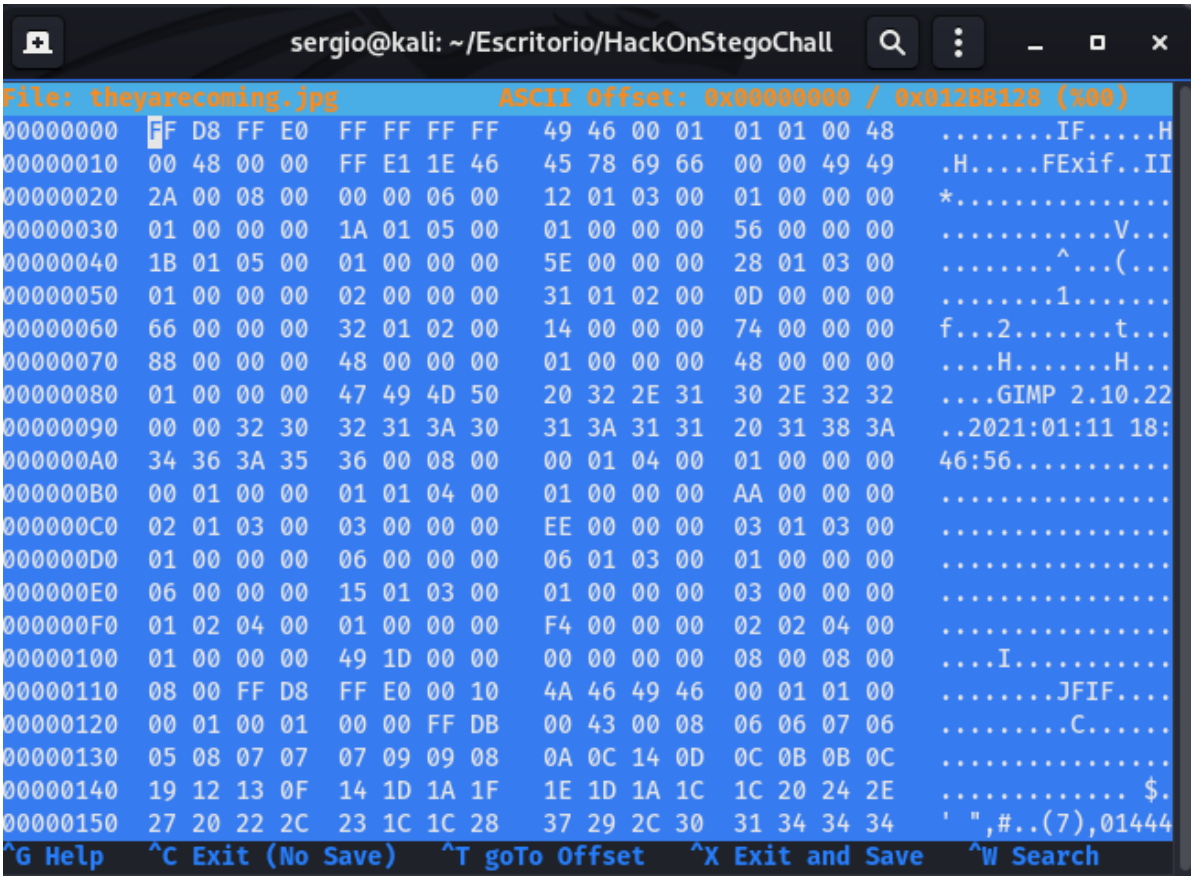
So, it seems clear that we have a corrupted file that we should fix to recover the image.

Let's go to the hexeditor

If we take a look into the file header with a hex editor:

```
sergio@kali:~/Escritorio/HackOnStegoChall$ hexeditor theyarecoming.jpg
```

And we remember the file signature of a JPG file (https://en.wikipedia.org/wiki/List_of_file_signatures) we clearly see that there is a problem with the file header:



So we correct the file header and...

Voilá! We can see the image!

Decoding the hidden message

The text in the image is clearly a Base64 encoded string, so we decode it using any tool like Cyberchef or the linux command line:

```
sergio@kali:~/Escritorio$ echo -n "cm9ja3lvdWlzeW91cmZyaWVuZA==" | base64 -d
```

Obtaining the string:

`rockyouisyourfriend`

It seems to suggest that we should to bruteforce the image to extract the hidden message...

Binwalk

At first, we can think that we must bruteforce the image, but if we check it using binwalk, we see something interesting:

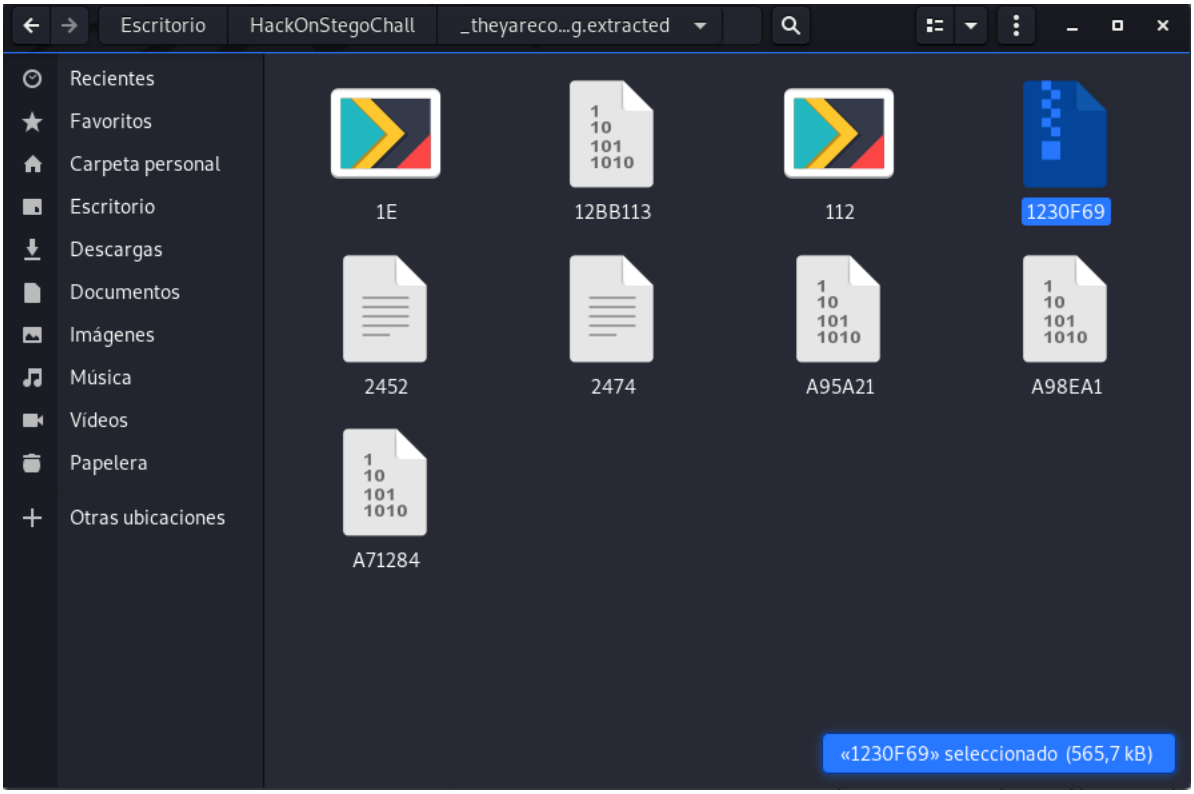
```
sergio@kali:~/Escritorio/HackOnStegoChall$ binwalk theyarecoming.jpg

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
30            0x1E            TIFF image data, little-endian offset of first
image directory: 8
274           0x112           JPEG image data, JFIF standard 1.01
9298          0x2452          Copyright string: "CopyrightOwner> <rdf:Seq/>
</plus:CopyrightOwner> <plus:Licensor> <rdf:Seq/> </plus:Licensor>
</rdf:Description> </rdf:RDF> </x:"
9332          0x2474          Copyright string: "CopyrightOwner> <plus:Licensor>
<rdf:Seq/> </plus:Licensor> </rdf:Description> </rdf:RDF> </x:xmpmeta>  "
10949252      0xA71284        Intel x86 or x64 microcode, sig 0x00000004, pf_mask
0x20000000, 2000-08-10, rev 0x-80000000, size 2048
11098657      0xA95A21        Intel x86 or x64 microcode, sig 0x09908000, pf_mask
0x2000100d, 2000-01-02, rev 0x0090, size 131072
11112097      0xA98EA1        Intel x86 or x64 microcode, sig 0x08000181, pf_mask
0x61000, 1A34-04-01, size 262160
19074921      0x1230F69       Zip archive data, encrypted compressed size:
565524, uncompressed size: 705644, name: flag.wav
19640595      0x12BB113       End of Zip archive, footer length: 22
```

We can see an encrypted Zip file in the image, which contains a flag.wav file inside. So we extract the zip file using binwalk:

`sergio@kali:~/Escritorio/HackOnStegoChall$ binwalk --dd='.*' theyarecoming.jpg`

And now we can try to crack its password



Time for cracking

Now, having obtained the clue of using rockyou in the first step of the challenge and the Zip file, we will try to crack it and extract the WAV file. To do this, we use John The Ripper and the zip2john utility to obtain the hash of the file.

`sergio@kali:~/Escritorio/HackOnStegoChall/_theyarecoming.jpg.extracted$ zip2john 1230F69 > ../ziphash`

And execute john with rockyou dictionary:

```
sergio@kali:~/Escritorio/HackOnStegoChall$ john ziphash --
wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (ZIP, WinZip [PBKDF2-SHA1 256/256 AVX2 8x])
Press 'q' or Ctrl-C to abort, almost any other key for status
forever          (passwordzip.zip/flag.wav)
1g 0:00:00:00 DONE (2021-01-11 19:44) 7.142g/s 914.2p/s 914.2c/s 914.2C/s
samantha..diamond
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

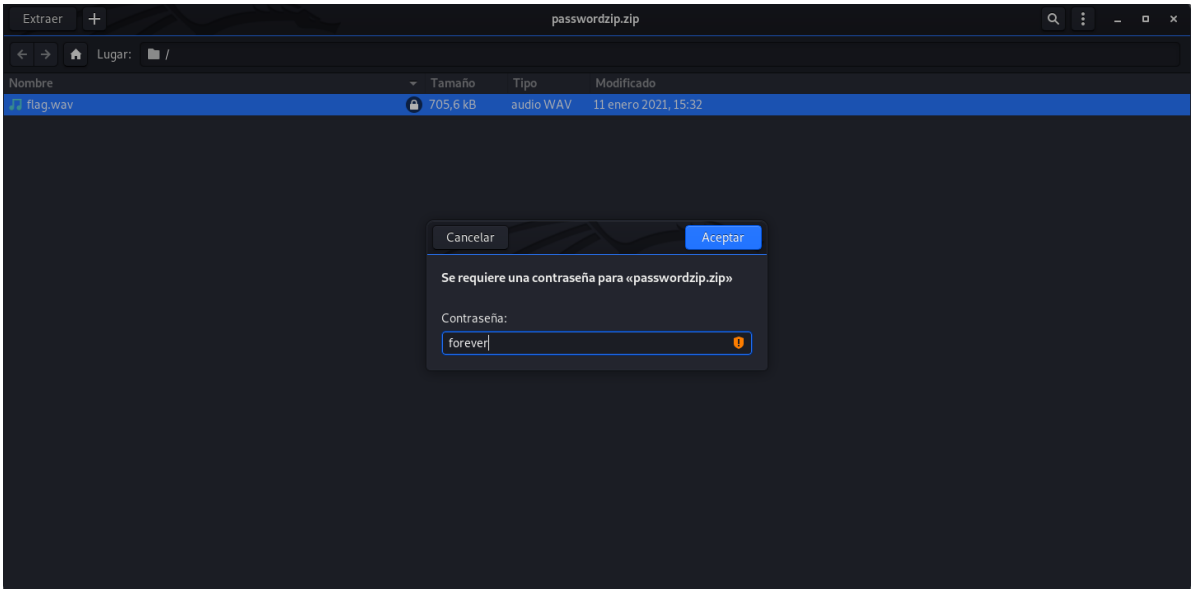
In few seconds/minutes, depending on your machine, we obtain the cracked password:

```
sergio@kali:~/Escritorio/HackOnStegoChall$ john ziphash --show
passwordzip.zip/flag.wav:forever:flag.wav:passwordzip.zip:passwordzip.zip

1 password hash cracked, 0 left
```

Extracting and... obtaining the flag?

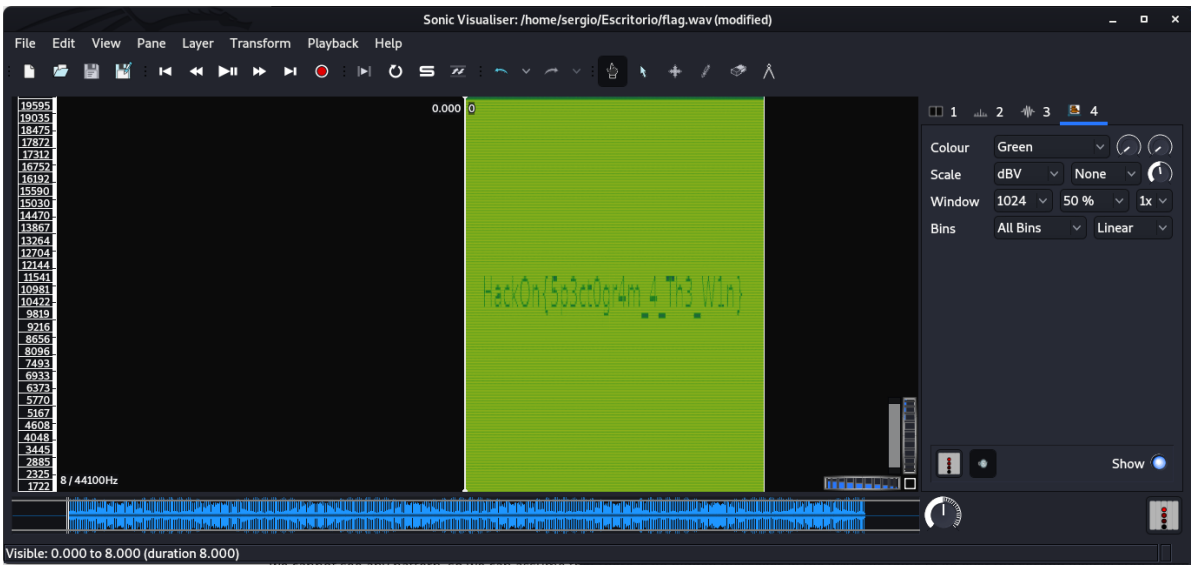
Using `forever` as password, we can extract the WAV file and if we listen it, we obtain the flag... or not.



If we open the file, we can not understand anything! Frustrating...

The spectogram is your friend

When we face with audio stego challenges, one of the first steps that we should take is to look into the wave of the file, looking for morse encoded messages, binary encoded messages, etc. For this task, we have tools like Audacity or Sonic visualiser. Analyzing the wave of the WAV file with Audacity, we cannot see any pattern, so we can assume that there is no hidden messages in the wave. Its tame to go over the second typicall step in audio stego challenges: analyze the spectogram. We do this with sonic visualiser and... BINGO!



FLAG: HackOn{5p3ct0gr4m_4_Th3_W1n}

Hope you have enjoyed this challenge and learned something :)