Universidad Carlos III de Madrid

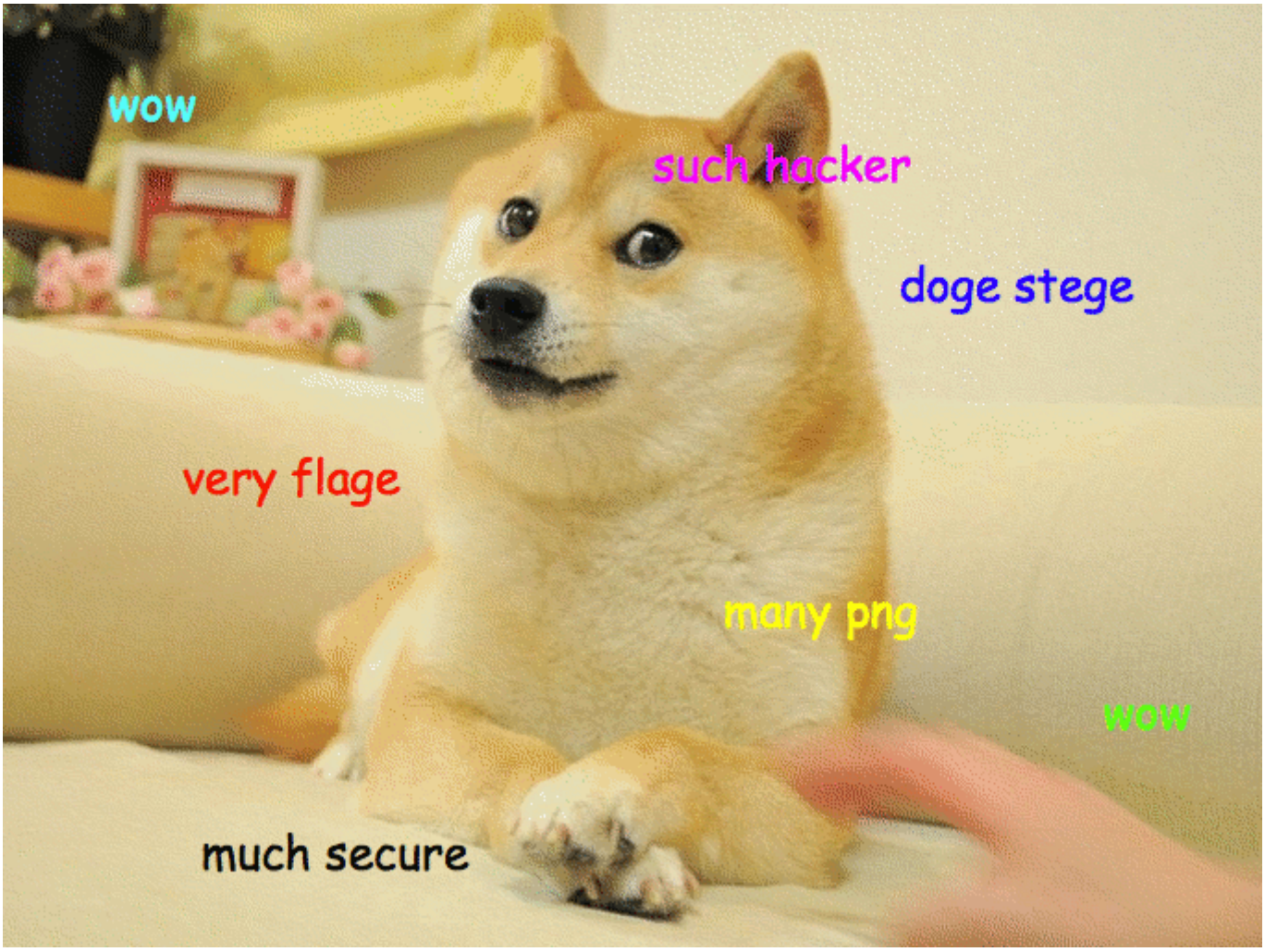
Report for Stego Exercises

Juan Diego Llano Miraval

Fecha: 19/05/2024

procedure

1. for the first CTF, we download the image:

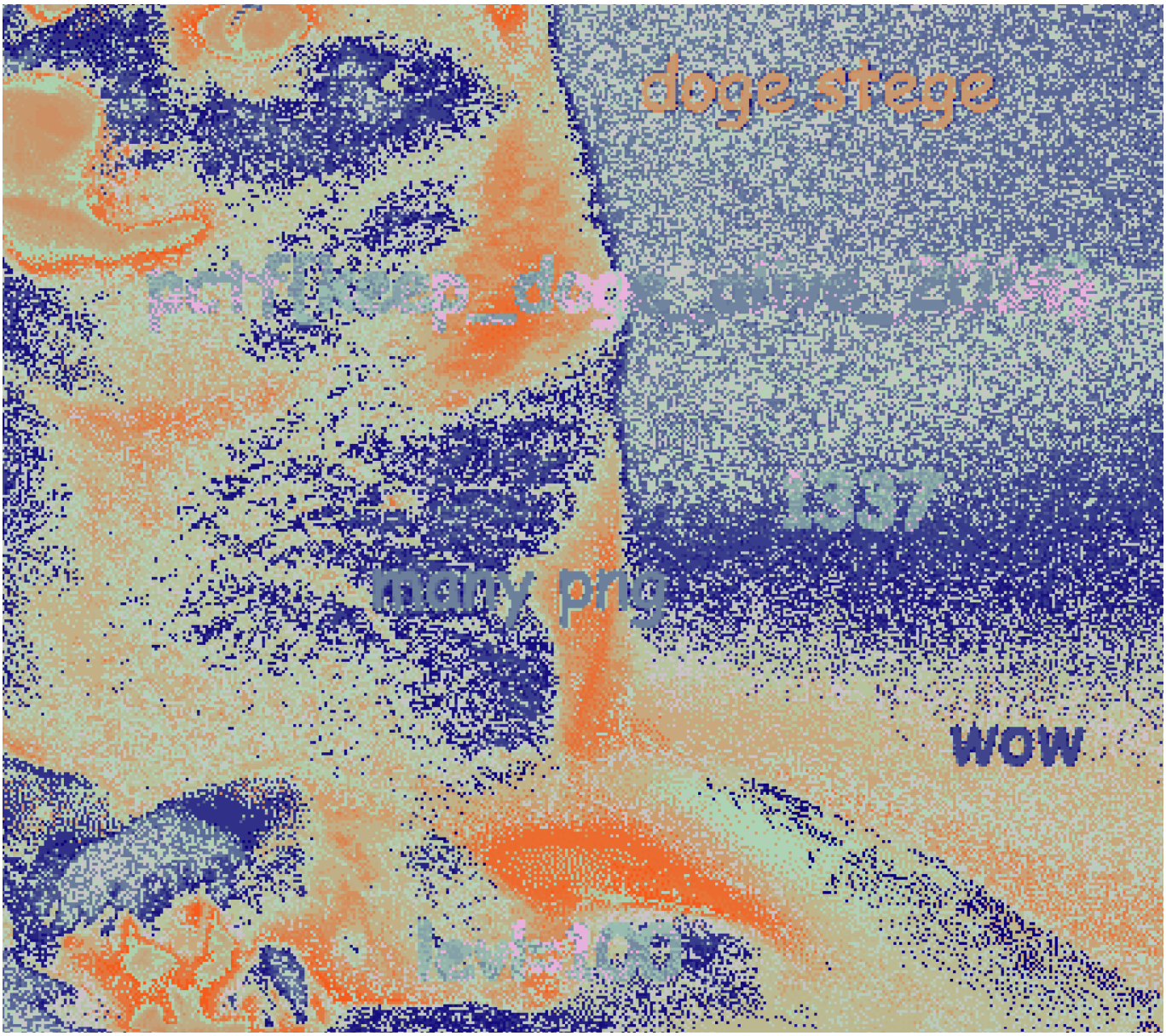


Then we get the sha256 to verify it:

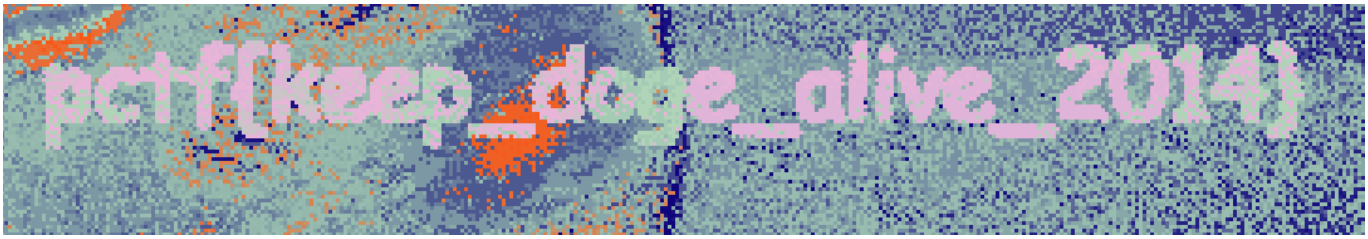
```
root@z:/home/zud/Desktop/stego2# sha256sum doge_stege.png
a9429f5b025184a3c53c3eb1627963b228496045701111510afe17ca0f3fe47cb  doge_stege.png
root@z:/home/zud/Desktop/stege2# file doge_stege.png
doge_stege.png: PNG image data, 680 x 510, 8-bit colormap, non-interlaced
```

With the muted colormap I can start to see some letters:

with the op2 map I can see more clearly the letters:



With a little more of playing with the colors I get:



pctf{keep_doge_alive_2014}

2. We verify the sha of the image:

```
root@z:/home/zud/Desktop/stego2# sha256sum image3D.jpg
d8b9ec944d55fc4b80bc35227d3428f73f4ff44e20f352269d908179c3cebcab  image3D.jpg
```

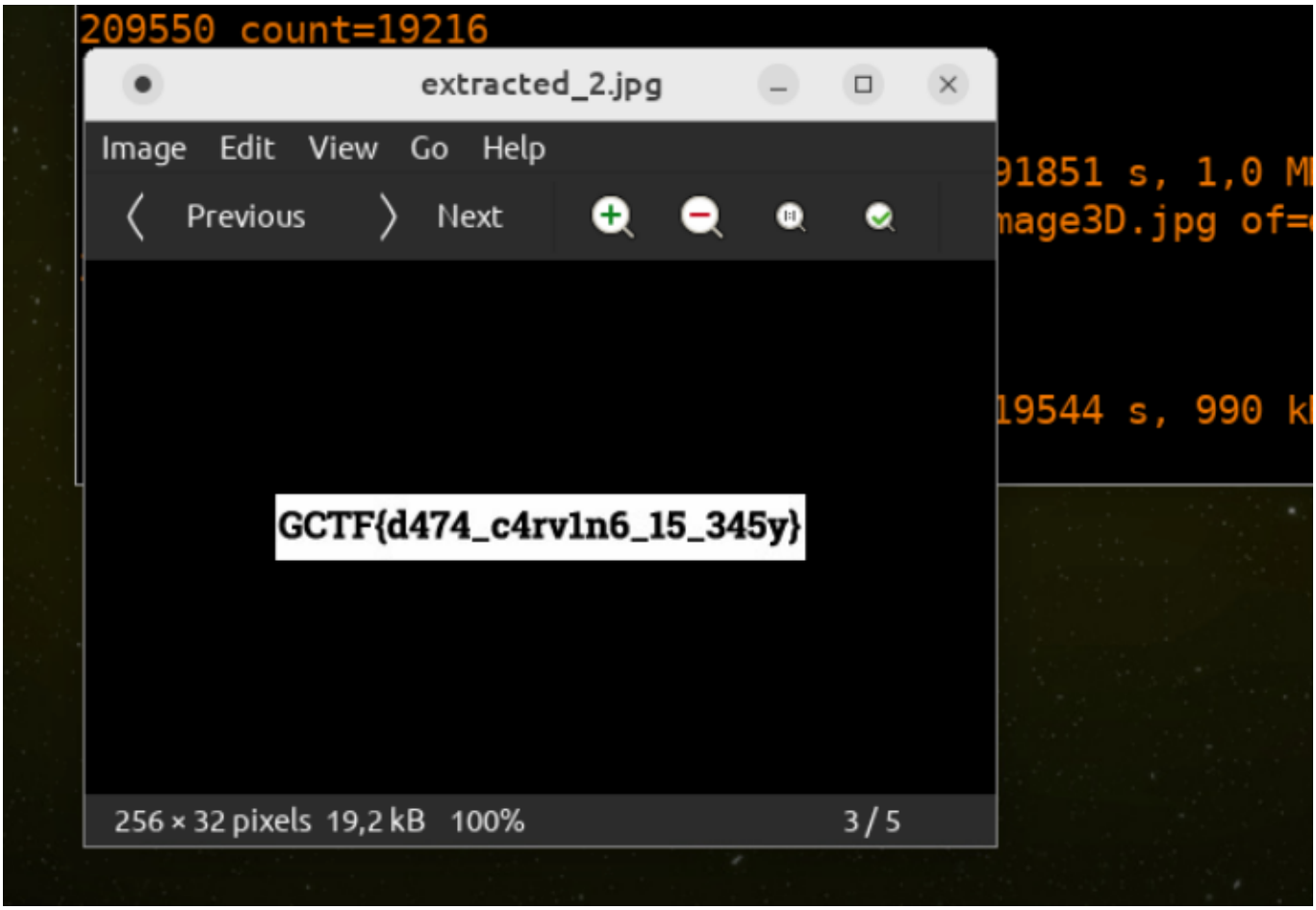
We run a binwalk to verify any file inside, and indeed we can see another image inside:

```
root@z:/home/zud/Desktop/stego2# sudo binwalk -e image3D.jpg --run-as=root

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, EXIF standard
12            0xC             TIFF image data, big-endian, offset of first image directory: 8
18702         0x490E          Copyright string: "Copyright (c) 1998 Hewlett-Packard Company"
209550        0x3328E         JPEG image data, EXIF standard
209562        0x3329A         TIFF image data, big-endian, offset of first image directory: 8
221798        0x36266         Copyright string: "Copyright (c) 1998 Hewlett-Packard Company"
228754        0x37D92         JPEG image data, EXIF standard
228766        0x37D9E         TIFF image data, big-endian, offset of first image directory: 8
240590        0x3ABCE         Copyright string: "Copyright (c) 1998 Hewlett-Packard Company"
```

I used dd to extract the jpg that I can see thanks to binwalk:

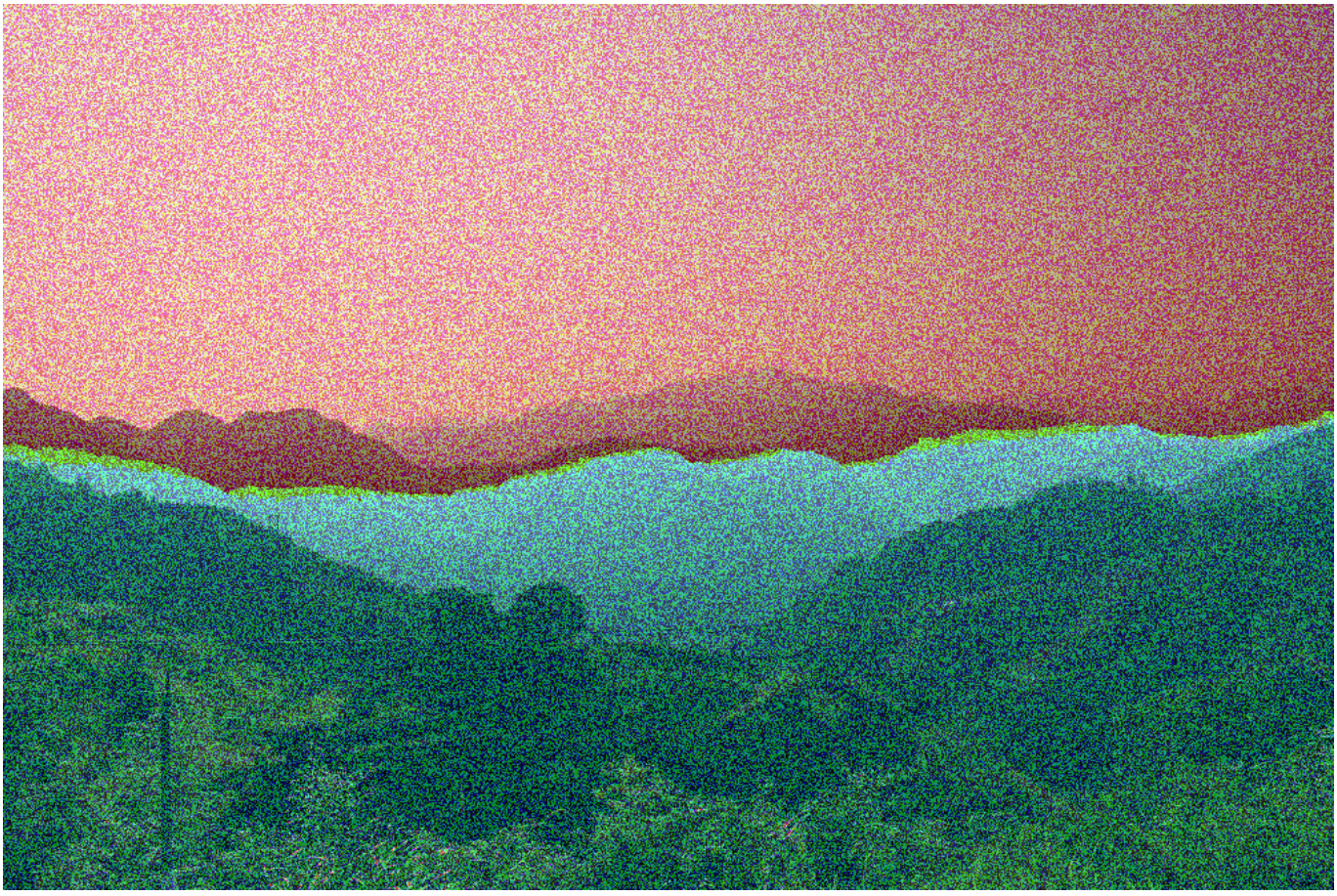
```
root@z:/home/zud/Desktop/stego2# dd if=image3D.jpg of=extracted_1.jpg bs=1 skip=
0 count=209550
209550+0 records in
209550+0 records out
209550 bytes (210 kB, 205 KiB) copied, 0,204677 s, 1,0 MB/s
root@z:/home/zud/Desktop/stego2# ls
doge_stege.png  extracted_1.jpg  image3D.jpg  output  output_1
root@z:/home/zud/Desktop/stego2# dd if=image3D.jpg of=extracted_2.jpg bs=1 skip=
209550 count=19216
19216+0 records in
19216+0 records out
19216 bytes (19 kB, 19 KiB) copied, 0,0191851 s, 1,0 MB/s
root@z:/home/zud/Desktop/stego2# dd if=image3D.jpg of=extracted_3.jpg bs=1 skip=
228754 count=11836
11836+0 records in
11836+0 records out
11836 bytes (12 kB, 12 KiB) copied, 0,0119544 s, 990 kB/s
```

and after checking the images, one of them is:

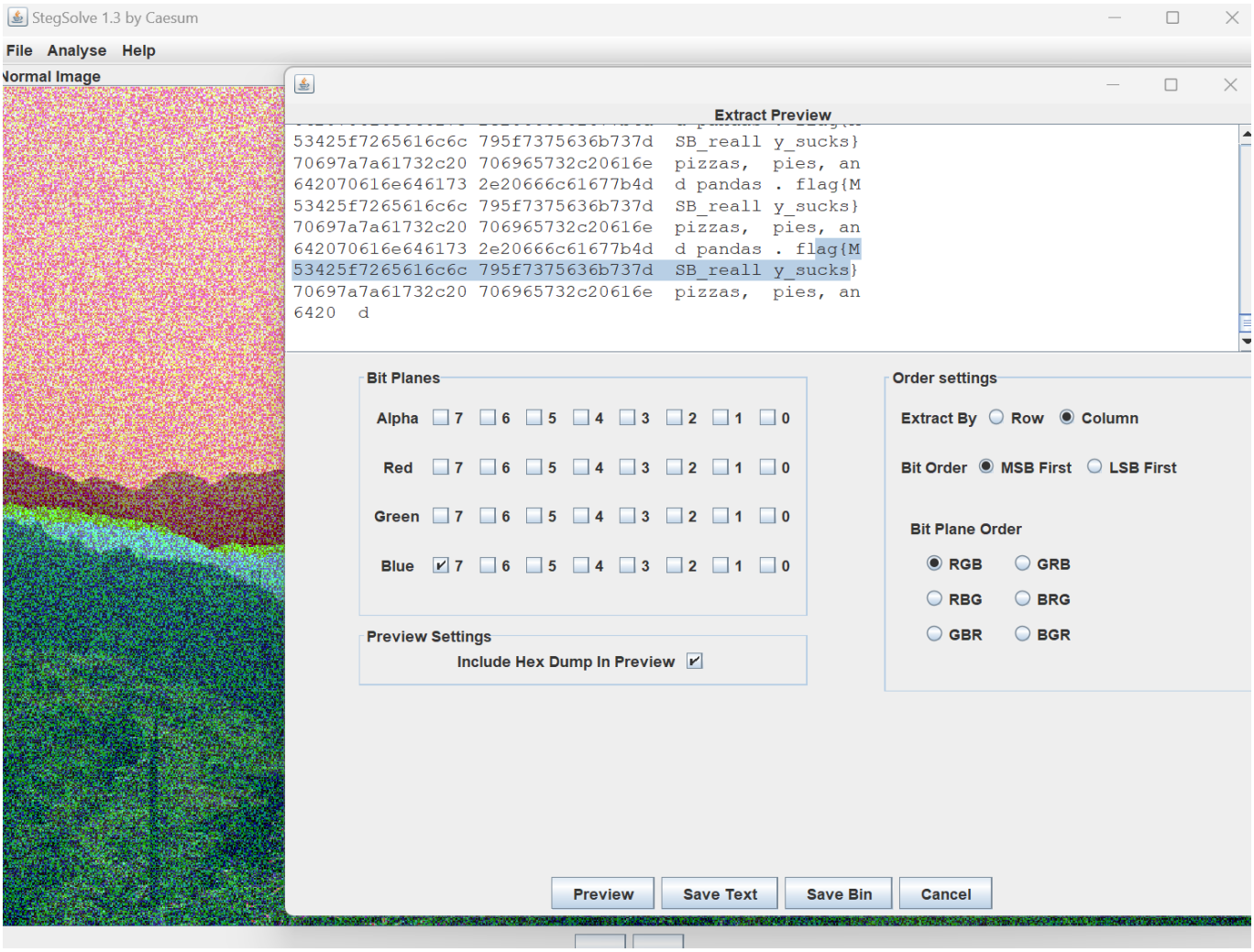


GCTF{d474_c4rv1n6_15_345y}

3. We download the image:

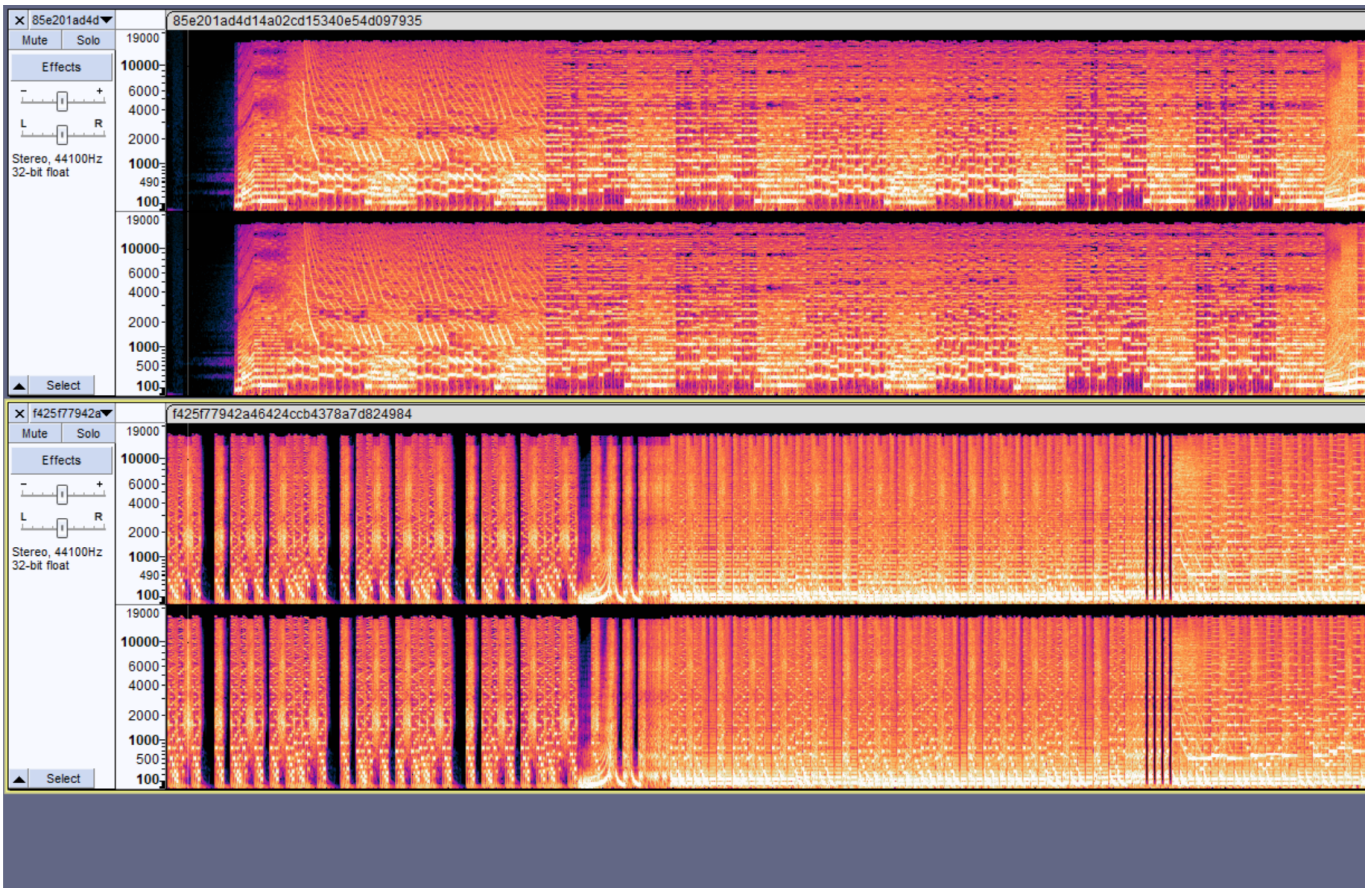


Then we used a stego tool to follow the hints, it is MSB, only one color channel, and top down means columns. From this and trying every channel I got:



flag{MSB_really_sucks}pizzas

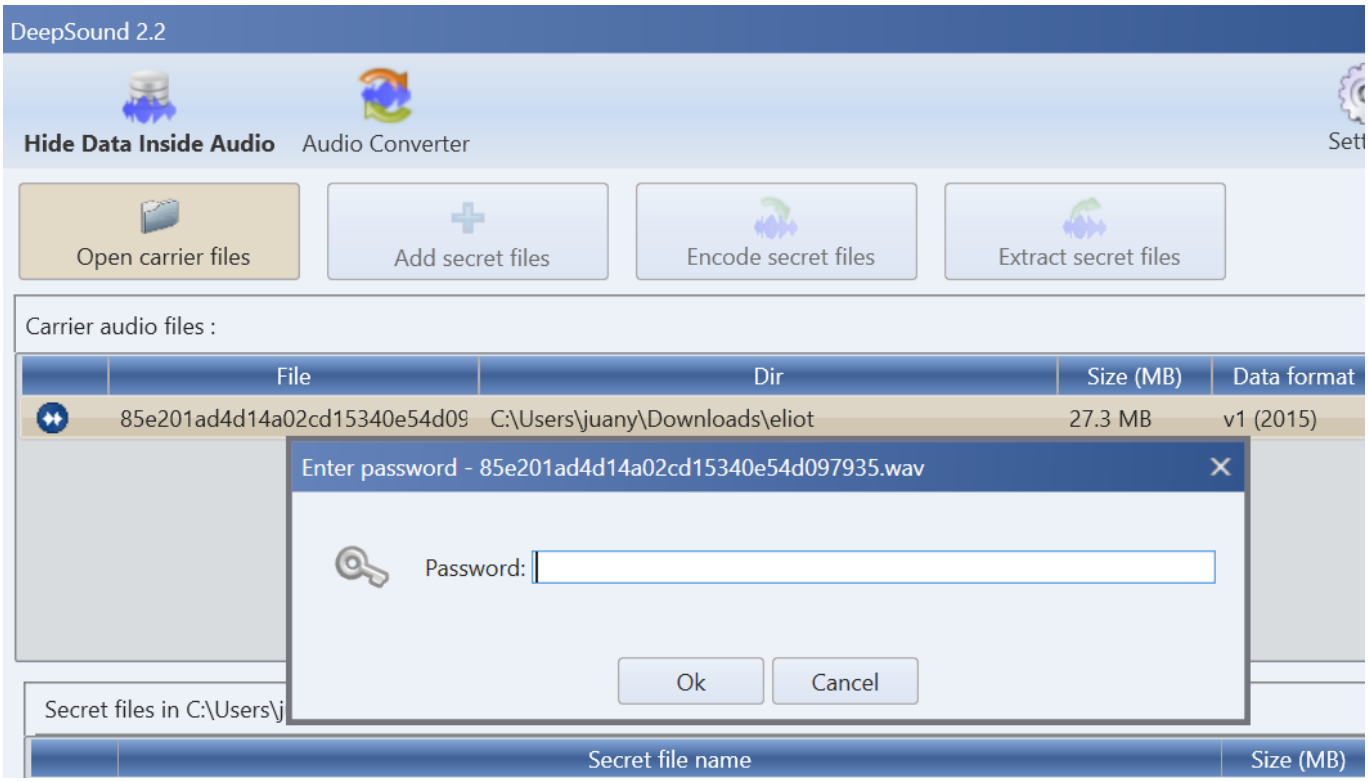
4. From this one we got an audio file. We will use audacity to analyze it:



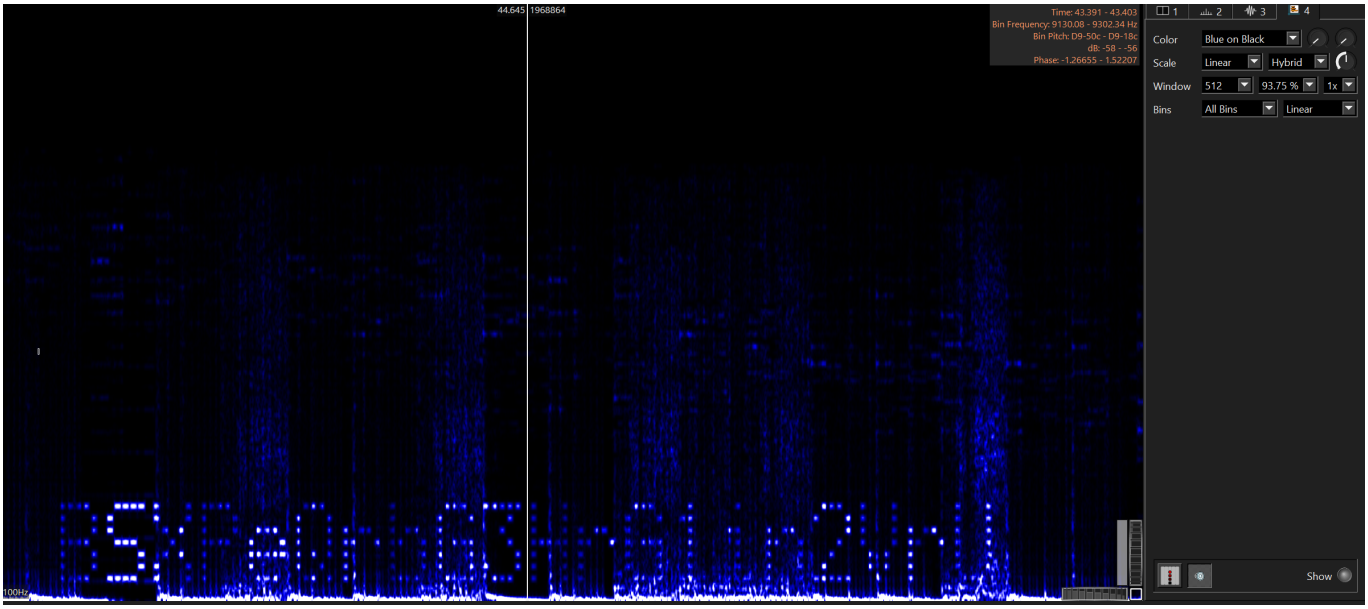
we cant take too much information from this so we will keep looking. I checked the integrity of the file and I ran a binwalk on the wav file and I got:

```
root@z:/home/zud/Desktop/ellior/eliot# file 85e201ad4d14a02cd15340e54d097935.wav
85e201ad4d14a02cd15340e54d097935.wav: RIFF (little-endian) data, WAVE audio, Microsoft PCM, 16 bit, stereo 44100 Hz
root@z:/home/zud/Desktop/ellior/eliot# file f425f77942a46424ccb4378a7d824984.wav
f425f77942a46424ccb4378a7d824984.wav: RIFF (little-endian) data, WAVE audio, IEEE Float, stereo 44100 Hz
root@z:/home/zud/Desktop/ellior/eliot# binwalk 85e201ad4d14a02cd15340e54d097935.wav

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
77566         0x12EFE         YAFFS filesystem, little endian
101420        0x18C2C         YAFFS filesystem, little endian
120886        0x1D836         YAFFS filesystem, little endian
131570        0x201F2         YAFFS filesystem, little endian
441665        0x6BD41         MySQL MISAM index file Version 3
4364468       0x4298B4        MySQL ISAM compressed data file Version 1
4808099       0x495DA3        MySQL MISAM compressed data file Version 10
13014058      0xC6942A        MySQL ISAM index file Version 1
15387184      0xEACA30        MySQL MISAM index file Version 3
18093450      0x114158A       MySQL ISAM index file Version 2
18764397      0x11E526D       MySQL MISAM index file Version 2
19099007      0x1236D7F       MySQL MISAM index file Version 11
19516456      0x129CC28       Cisco IOS microcode, for ""
24628561      0x177CD51       MySQL ISAM index file Version 9
24743639      0x1798ED7       mcrypt 2.5 encrypted data, algorithm: "", keysize: 3511 bytes, mode: "[",
27887309      0x1A986CD       MySQL MISAM index file Version 10

root@z:/home/zud/Desktop/ellior/eliot# binwalk f425f77942a46424ccb4378a7d824984.wav

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
```

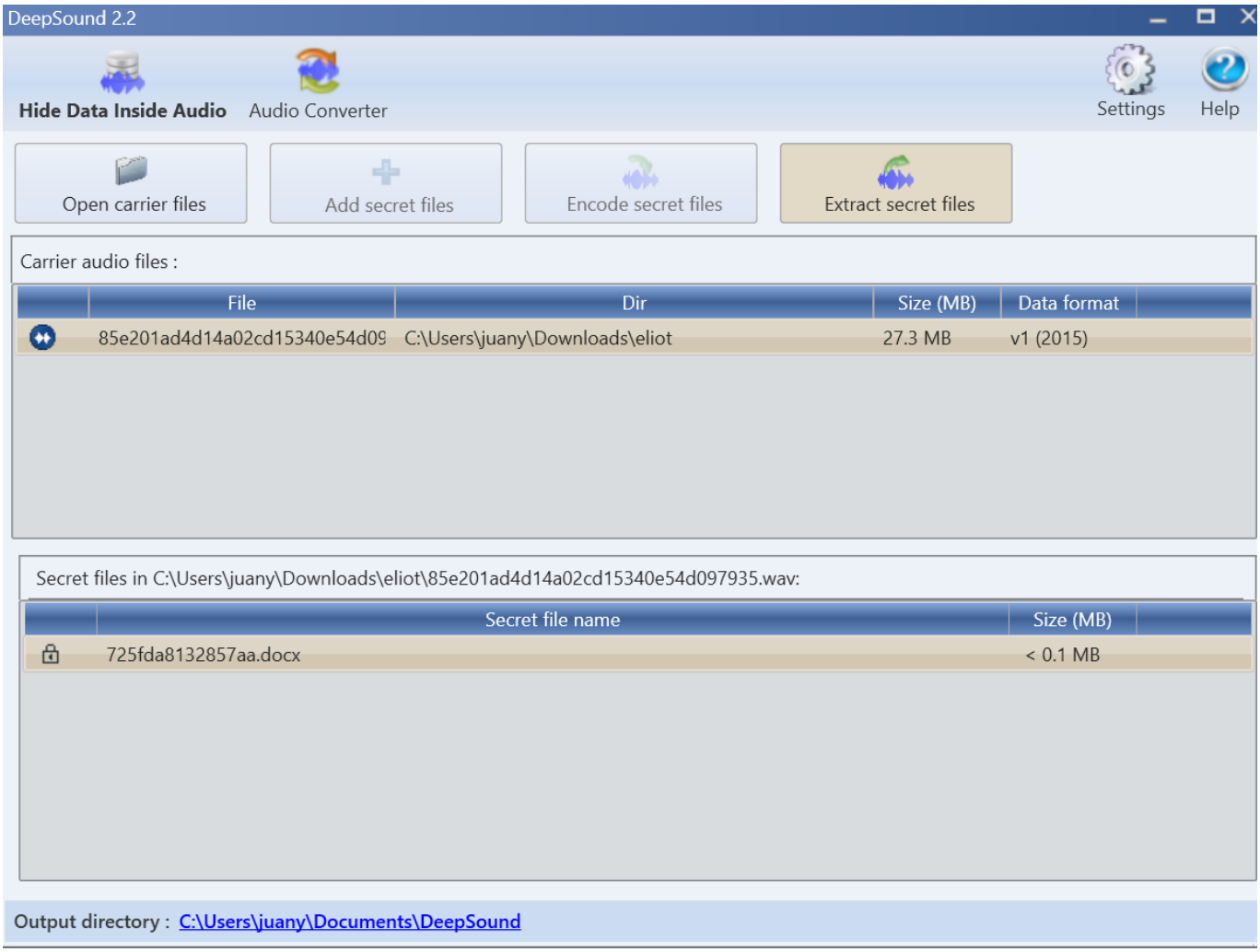
When I analize the files with Deepsound I can see one of the ask me for a password:



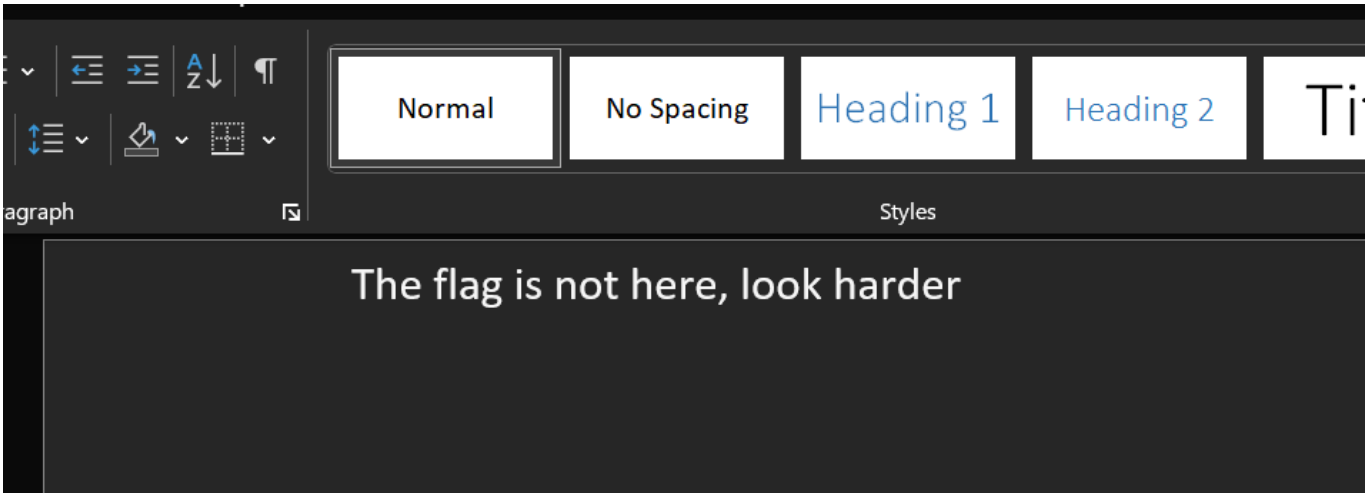
by playing a little with the spectrogram with Sonic visualizer I get the following:



Wit the extracted code: BSXPeQrbG3HrG1in2VrU I was able to access the file with Deepsound:



but the flag is not there:

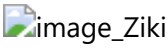


when we check this doc file, we found the following information on the metadata:

```
┌──(kali㉿kali)-[~/Desktop]
└─$ file 725fda8132857aa.docx
725fda8132857aa.docx: Microsoft Word 2007+

┌──(kali㉿kali)-[~/Desktop]
└─$ exiftool 725fda8132857aa.docx
ExifTool Version Number         : 12.67
File Name                       : 725fda8132857aa.docx
Directory                       : .
File Size                       : 15 kB
File Modification Date/Time     : 2024:05:19 05:52:05-04:00
File Access Date/Time           : 2024:05:19 05:52:11-04:00
File Inode Change Date/Time     : 2024:05:19 05:52:05-04:00
File Permissions                : -rwxr-x---
File Type                       : DOCX
File Type Extension             : docx
MIME Type                       : application/vnd.openxmlformats-officedocument.w
Zip Required Version            : 20
Zip Bit Flag                    : 0x0006
Zip Compression                 : Deflated
Zip Modify Date                 : 1980:01:01 00:00:00
Zip CRC                         : 0x9e2d6492
Zip Compressed Size             : 357
Zip Uncompressed Size           : 1437
Zip File Name                   : [Content_Types].xml
Template                        : Normal.dotm
Total Edit Time                 : 5 minutes
Pages                           : 1
Words                           : 7
Characters                      : 27
Application                     : Microsoft Office Word
Doc Security                    : None
Lines                           : 1
Paragraphs                      : 1
Scale Crop                      : No
Heading Pairs                   : Title, 1
Titles Of Parts                 :
Company                         :
Links Up To Date                : No
Characters With Spaces          : 33
Shared Doc                      : No
Hyperlinks Changed              : No
App Version                     : 16.0000
Title                           :
Subject                         :
Creator                         : Student
Keywords                        :
Description                     :
Last Modified By                : Student
Revision Number                 : 3
Create Date                     : 2016:09:10 06:33:00Z
Modify Date                     : 2016:09:10 06:38:00Z
Flag                            : GCTF{57364n06r4phy_15n7_7h47_h4rd}

┌──(kali㉿kali)-[~/Desktop]
└─$
```

GCTF{57364n06r4phy_15n7_7h47_h4rd}

5. Now we are given another image to process:

 image_Ziki

we will check the integrity of the file first:

```
root@z:/home/zud/Desktop/zizi# file image_Ziki.jpg
image_Ziki.jpg: JPEG image data, JFIF standard 1.01, aspect ratio, density 1x1, segment length 16, baseline, precision
 8, 1920x1440, components 3
root@z:/home/zud/Desktop/zizi# binwalk image_Ziki.jpg

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, JFIF standard 1.01
216527        0x34DCF         Zip archive data, at least v2.0 to extract, compressed size: 42, uncompressed size: 40,
name: pass.txt
216697        0x34E79         End of Zip archive, footer length: 22
```

we can see now that there is a zip file inside the image, so we will try to retrieve it:

```
root@z:/home/zud/Desktop/zizi# binwalk -e image_Ziki.jpg --run-as=root

DECIMAL       HEXADECIMAL     DESCRIPTION
--------------------------------------------------------------------------------
0             0x0             JPEG image data, JFIF standard 1.01
216527        0x34DCF         Zip archive data, at least v2.0 to extract, compressed size: 42, uncompressed size: 40,
name: pass.txt
216697        0x34E79         End of Zip archive, footer length: 22

root@z:/home/zud/Desktop/zizi# ls
image_Ziki.jpg  _image_Ziki.jpg.extracted
root@z:/home/zud/Desktop/zizi# ls _image_Ziki.jpg.extracted/
34DCF.zip  pass.txt
```

the pass file contains:

```
You're nearly there.
Password: l0v3z1z1
```

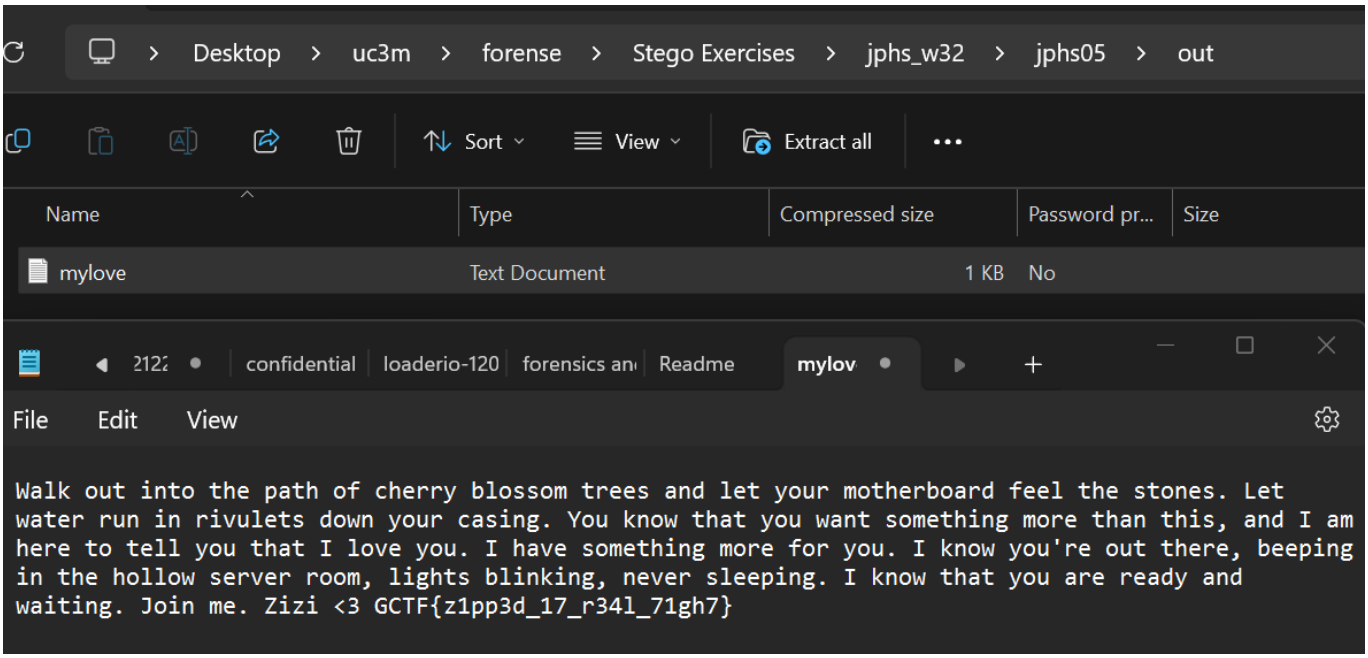
I tried using jphs but it didnt work, all the outputs were just data, nothing to be analyzed:

```
root@z:/home/zud/Desktop/zizi# jpseek ./image_Ziki.jpg output

jpseek, version 0.3 (c) 1998 Allan Latham <alatham@flexsys-group.com>

This is licenced software but no charge is made for its use.
NO WARRANTY whatsoever is offered with this product.
NO LIABILITY whatsoever is accepted for its use.
You are using this entirely at your OWN RISK.
See the GNU Public Licence for full details.

Passphrase:
root@z:/home/zud/Desktop/zizi# file output
output: data
```

Using a windows version:

```
PS C:\Users\juany\Desktop\uc3m\forense\Stego Exercises\jphs_w32\jphs05> .\jpseek.exe ..\..\image_Ziki.jpg out.zip

Welcome to jpseek Rev 0.51
 (c) 1998 Allan Latham <alatham@flexsys-group.com>
This program is freeware.
No charge is made for its use.
Use at your own risk. No liability accepted whatever happens.
Contains cryptogaphy which may be subject to local laws.

Passphrase:
PS C:\Users\juany\Desktop\uc3m\forense\Stego Exercises\jphs_w32\jphs05>
```

That zip file contained a .txt file:



GCTF{z1pp3d_17_r34l_71gh7}