INSTITUTO SUPERIOR TÉCNICO

Departamento de Engenharia Informática

Forensics Cyber Security

MEIC / METI 2024-2025 – 1st Period

Digital Forensics Report Lab1

Group number:	G43	Name	IST Number
Student 1:		Daniel Pereira	99194
Student 2:		Luana Marques	82374
Student 3:		Sofia Du	104195

1 Acquired artifacts

Name	Type	SHA-256 Value
rocket.png	png	28994ee87128ab44aa51ab82161507418070bef6708d65014327140c34f3f747
api_message.txt	txt	03566817f143262cd35c6c518ab303bd3127c62f2f011301fba9e227be438ecf
nmap_passwords.txt	txt	bfe6869955ed21e77bb510ea140b00ae2986cd6d08c7bf6f99c8aa2b8d20755a
wallpaper_extracted_file.pdf	pdf	04c099abe319fe6dd90c420161b64523c0332905332640213eceea3f205500c8
best-intro.pdf	pdf	6ac9c4d35d790a3a2d3f3b3b1866a2f5d0e40c08a839799c39643c238e705829
blank.pdf	pdf	f5fed0c14eb7462c145c2515ca40429b790b16b0e42671ae06c4fdfda3bd94a7

2 Report of all findings

The initial step of this investigation involved checking the fingerprints of each file, using ***sha256sum file_name***. By doing this we confirmed that our files had not been manipulated and were the same as the ones student Musk had. We analyzed all files in detail and found 6 secrets within the provided files.

<u>secret 1</u> (api_message.txt): andromeda.png & cartwheel.tiff & lactea.jpg

First, we checked them with the command file to see if there is anything unusual about these images. Then we used the exiftool command (***exiftool file_name***) for reading, writing, and manipulating image, audio, video, etc.

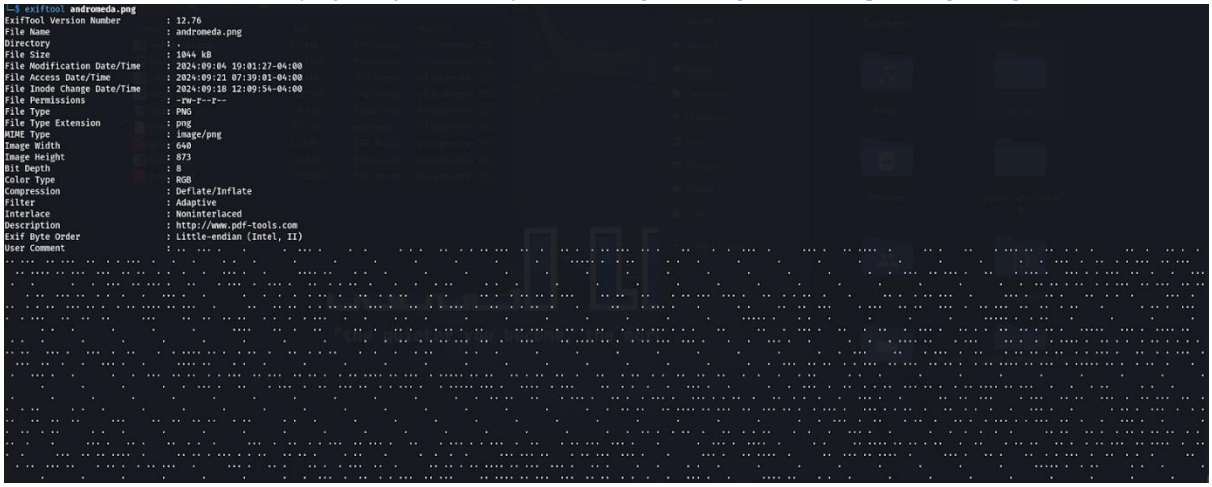


Figure 1 - andromeda metadata

In the User Comment section, we can see some type of hidden message in a series of dots and spaces in the andromeda, cartwheel and lacteal images. After dismissing the dots and spaces as some kind of morse code, we created a python script to turn these hidden messages into zeroes and ones, and then into strings.

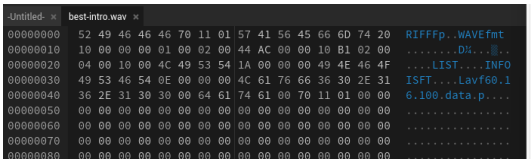
```python
def dots_to_bits(dot_string):
    # Convert dots and spaces to binary string
    binary_string = dot_string.replace('.', '1').replace(' ', '0')
    return binary_string

def binary_to_string(binary_string):
    # Split the binary string into 8-bit chunks and convert to characters
    chars = []
    for i in range(0, len(binary_string), 8):
        byte = binary_string[i:i+8]
        if len(byte) == 8:  # Ensure it's a complete byte
            chars.append(chr(int(byte, 2)))  # Convert binary to integer, then to character
    return ''.join(chars)
```

Figure 2 - script to turn dots to bits

After running the script with the 3 files, only lactea gave a readable Json string, but looked like it was cut.

After appending the bits of the lactea, andromeda and cartwheel in that order, we got the full secret message.

secret 2 (best-intro.pdf): best-intro.wav



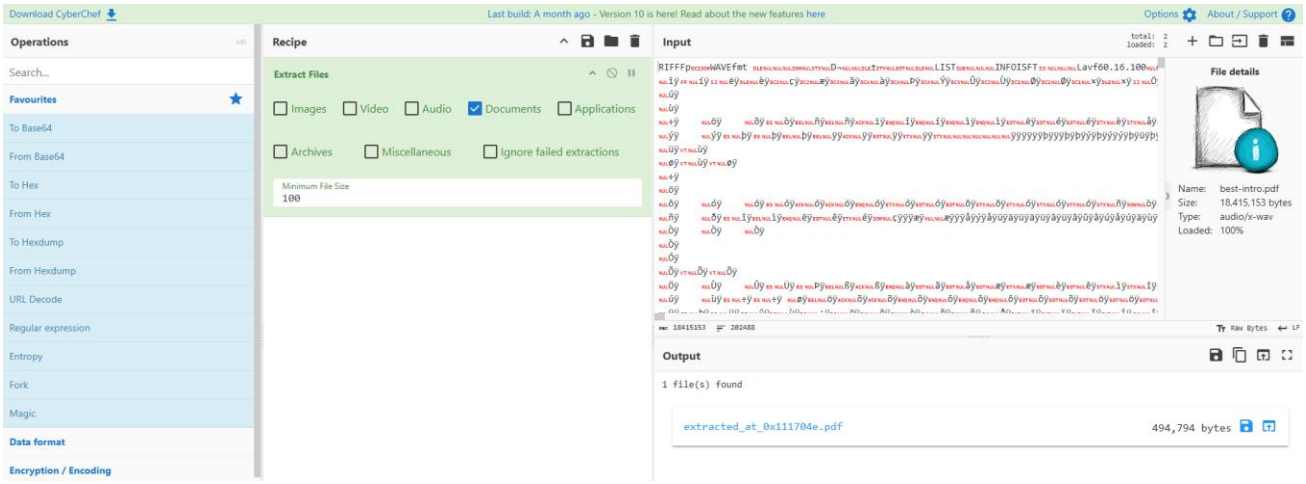
38 42 50 53	8BPS	0	psd	Document file, Adobe Photosh native file forma
52 49 46 46 ?? ?? ?? ?? 57 41 56 45	RIFF????WAVE	0	wav	Waveform Audi File Format[36]

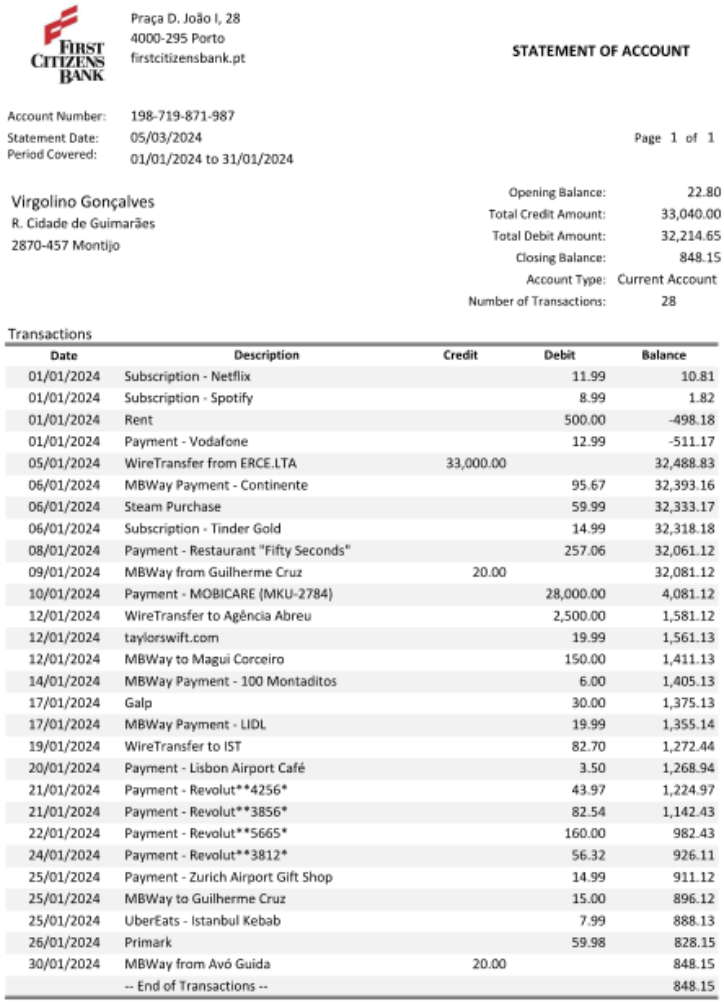
Figure 3 - magic numbers

we started by using the command ***hexdump –c best-intro.wav | head –n 10*** and the HexEd.it website (HexEd.it - Browser-based Online and Offline Hex Editing) to obtain the first few bytes of the file in search of its magic numbers. The output showed us that "52 49 46 46 70 11 01 57 41 56 45" corresponds to a magic number of the WAV file type, as verified by searching for it in the List of file signatures - Wikipedia. Everything seemed fine, but when we used another command ***hexdump –c best-intro.wav | tail –n 10*** to analyze the last few bytes of the file, we saw that it

had magic numbers that are related to a PDF file format. We simply added the ".pdf" extension to the file and attempted to open the hidden PDF file. (best-intro.pdf - Virgolino´s statement of account)

After noticing that only changing the file extension to pdf made it unable to open in windows (in linux it works fine) and also was too big, so we used the CyberChef tool online to extract the pdf, showing that the hidden pdf file only starts at the address 0x111704e





FIRST CITIZENS BANK

Praça D. João I, 28
4000-295 Porto
firstcitizensbank.pt

STATEMENT OF ACCOUNT

Account Number: 198-719-871-987
Statement Date: 05/03/2024
Period Covered: 01/01/2024 to 31/01/2024

Page 1 of 1

Virgolino Gonçalves
R. Cidade de Guimarães
2870-457 Montijo

Opening Balance:	22.80
Total Credit Amount:	33,040.00
Total Debit Amount:	32,214.65
Closing Balance:	848.15
Account Type:	Current Account
Number of Transactions:	28

Transactions

Date	Description	Credit	Debit	Balance
01/01/2024	Subscription - Netflix		11.99	10.81
01/01/2024	Subscription - Spotify		8.99	1.82
01/01/2024	Rent		500.00	-498.18
01/01/2024	Payment - Vodafone		12.99	-511.17
05/01/2024	WireTransfer from ERCE.LTA	33,000.00		32,488.83
06/01/2024	MBWay Payment - Continente		95.67	32,393.16
06/01/2024	Steam Purchase		59.99	32,333.17
06/01/2024	Subscription - Tinder Gold		14.99	32,318.18
08/01/2024	Payment - Restaurant "Fifty Seconds"		257.06	32,061.12
09/01/2024	MBWay from Guilherme Cruz	20.00		32,081.12
10/01/2024	Payment - MOBICARE (MKU-2784)		28,000.00	4,081.12
12/01/2024	WireTransfer to Agência Abreu		2,500.00	1,581.12
12/01/2024	taylorswift.com		19.99	1,561.13
12/01/2024	MBWay to Magui Corceiro		150.00	1,411.13
14/01/2024	MBWay Payment - 100 Montaditos		6.00	1,405.13
17/01/2024	Galp		30.00	1,375.13
17/01/2024	MBWay Payment - LIDL		19.99	1,355.14
19/01/2024	WireTransfer to IST		82.70	1,272.44
20/01/2024	Payment - Lisbon Airport Café		3.50	1,268.94
21/01/2024	Payment - Revolut**4256*		43.97	1,224.97
21/01/2024	Payment - Revolut**3856*		82.54	1,142.43
22/01/2024	Payment - Revolut**5665*		160.00	982.43
24/01/2024	Payment - Revolut**3812*		56.32	926.11
25/01/2024	Payment - Zurich Airport Gift Shop		14.99	911.12
25/01/2024	MBWay to Guilherme Cruz		15.00	896.12
25/01/2024	UberEats - Istanbul Kebab		7.99	888.13
26/01/2024	Primark		59.98	828.15
30/01/2024	MBWay from Avó Guida	20.00		848.15
	-- End of Transactions --			848.15

Figure 4a – bank statement

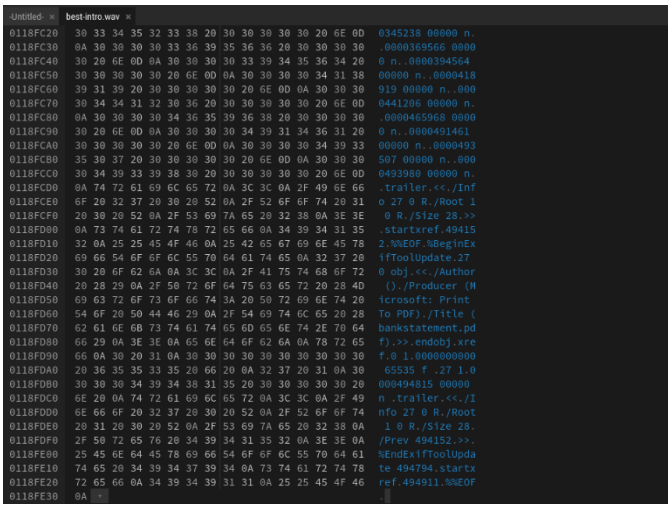


Figure 4b – last few bytes of the best-intro.wav file

secret 3 (wallpaper_extracted_file.pdf) & secret 4 (blank.pdf): myzip.zip

Wallpaper.png, hd.jpg, got.jpg and blank, these files were discovered by cracking the password of the myzip.zip file. As expected, we began the analysis of this Zip file by attempting to open it and extract its contents. However, we immediately discovered that it was protected by a password. To successfully extract the ZIP file´s content, we would have to somehow discover the password.

First, we use John the Ripper with the *rockyou* wordlist to launch a brute-force attack against the zip file. This was done through two commands:

- ***zip2john myzip.zip > hash.txt***
- ***john --wordlist=/usr/share/wordlists/rockyou.txt hash.txt***

Unfortunately, we didn´t find any possible password. We started to notice that the file thrones.pdf contains random words, maybe it's a story. So, we decided to create our wordlist with this file's content. We then used John the Ripper with the following commands:

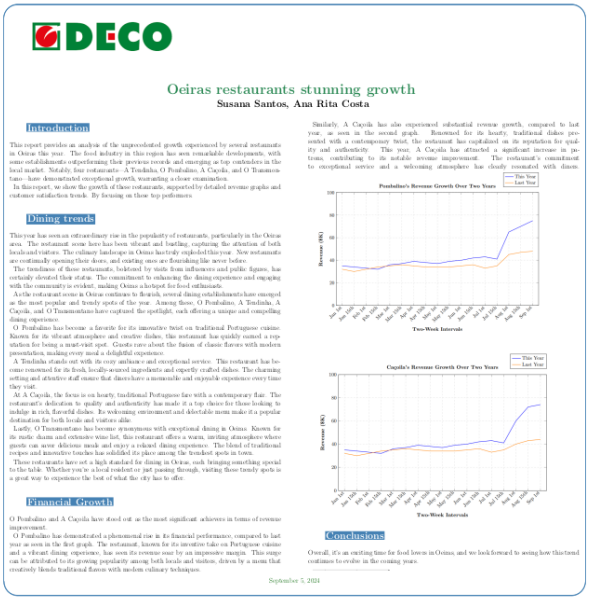
- ***zip2john myzip.zip > hash.txt***
- ***john --wordlist=output_words.txt hash.txt***

```
┌──(kali㉿kali)-[~/csf-lab1-artifacts]
└─$ john --wordlist=output_words.txt hash.txt
Using default input encoding: UTF-8
Loaded 1 password hash (PKZIP [32/64])
Will run 3 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
zmaistangeptotlargelynaejotseda (myzip.zip)
1g 0:00:00:00 DONE (2024-09-19 10:41) 33.33g/s 204800p/s 204800c/s 204800C/s Se..ks
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Figure 5 - zip password

```python
import PyPDF2
import re

def extract_words_from_pdf(pdf_path, output_txt_path):
    with open(pdf_path, 'rb') as pdf_file:
        reader = PyPDF2.PdfReader(pdf_file)
        all_text = ""

        # Extract text from each page of the PDF
        for page_num in range(len(reader.pages)):
            page = reader.pages[page_num]
            all_text += page.extract_text()

        # Use regex to find all words
        words = re.findall(r'\b\w+\b', all_text)

        # Write words, each on a new line
        with open(output_txt_path, 'w') as output_file:
            for word in words:
                output_file.write(word + '\n')

# Usage
pdf_path = "thrones.pdf"  # Replace with your PDF file path
output_txt_path = "output_words.txt"  # Replace with your desired output text file path
extract_words_from_pdf(pdf_path, output_txt_path)
```

Figure 6 - script to create wordlist from thrones pdf



DECO

Oeiras restaurants stunning growth

Susana Santos, Ana Rita Costa

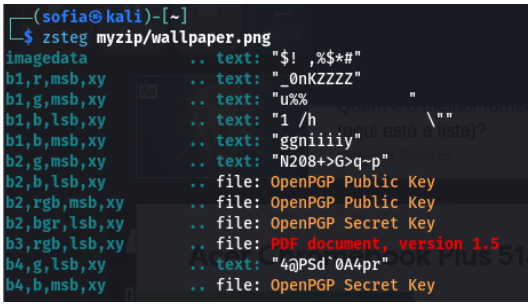
Introduction

Dining trends

Financial Growth

Conclusions

Figure 7 - restaurant growth report



```
┌──(sofia㉿kali)-[~]
└─$ zsteg myzip/wallpaper.png
imagedata           .. text: "$! ,%$*#"
b1,r,msb,xy         .. text: "_0nKZZZZ"
b1,g,msb,xy         .. text: "u%%                "
b1,b,lsb,xy         .. text: "1 /h                \""
b1,b,msb,xy         .. text: "ggniiiiy"
b2,g,msb,xy         .. text: "N208+>G>q~p"
b2,b,lsb,xy         .. file: OpenPGP Public Key
b2,rgb,msb,xy       .. file: OpenPGP Public Key
b2,bgr,lsb,xy       .. file: OpenPGP Secret Key
b3,rgb,lsb,xy       .. file: PDF document, version 1.5
b4,g,lsb,xy         .. text: "4@PSd`0A4pr"
b4,b,msb,xy         .. file: OpenPGP Secret Key
```

Figure 8 - zsteg command

Having found the password for the ZIP file (**password: zmaistangeptotlargelynaejotseda)**, we were finally able to analyze the hidden files, which allowed us to get more context on the crime being committed. The third secret we found here was a hidden PDF document embedded in the wallpaper.png file. After analyzing its magic numbers and relevant strings using the command ***strings -n 8 wallpaper.png,*** we used the **zsteg** command and detected a stegano-hidden data. This revealed a PDF file about Oeiras Restaurants.

```
┌──(sofia㉿kali)-[~]
└─$ zsteg -E b3,rgb,lsb,xy  myzip/wallpaper.png > wallpaper_extracted_file

┌──(sofia㉿kali)-[~]
└─$ file wallpaper_extracted_file
wallpaper_extracted_file: PDF document, version 1.5
```

Figure 9 – Extract PDF

The fourth secret was found in a file named "blank." Since this file did not have a file type, we began our investigation using the **hexed.it** website to gather information about its file type. The output showed us that the first bytes of the file were "25 50 4E 47 2D." After obtaining this byte sequence, we checked if it corresponded to any known file type's magic numbers by searching for it on the "List of file signatures" Wikipedia page. There, we discovered that these bytes correspond to the magic numbers for a PDF file. Finally, we simply added the ".pdf" extension to the file and attempted to open it. Inside, we found documentation related to the MKUltra Mind Control Component API.

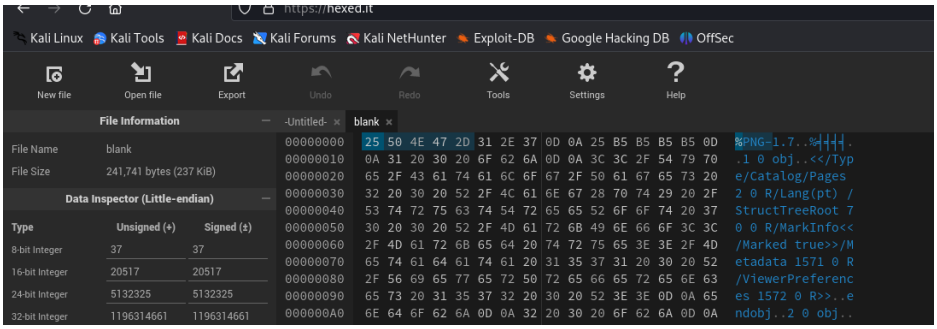


Figure 10 - hex editor

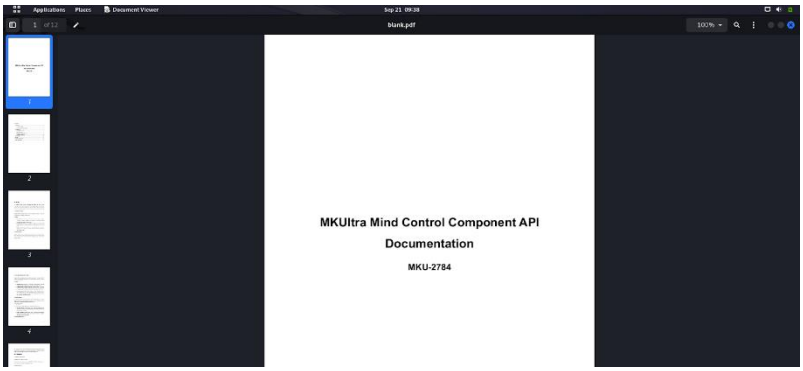


Figure 11 - MKUltra PDF

secret 5(nmap_passwords.txt): nmap

We first found the magic numbers of the file with ***hexdump -c nmap | head -n 10***, which is 7F 45 4C 46, an Executable and Linking Format (ELF). With the file command, we got missing section headers.

```
┌──(sofia㉿kali)-[~]
└─$ file nmap.elf
nmap.elf: ELF 64-bit LSB shared object, x86-64, version 1 (GNU/Linux), statically linked, missing section headers at 2866672
```

Figure 12 - nmap file command output

When viewing the file contents in a text editor, we noticed that only the header is in the ELF format, the rest appears to be encoded in base64 as noticed by the equal character in the end, a known padding in that encoding format.

We used the tool CyberChef to decode the base64 of the file, removing first the ELF header and getting a list of usernames and passwords, as shown in the picture below.

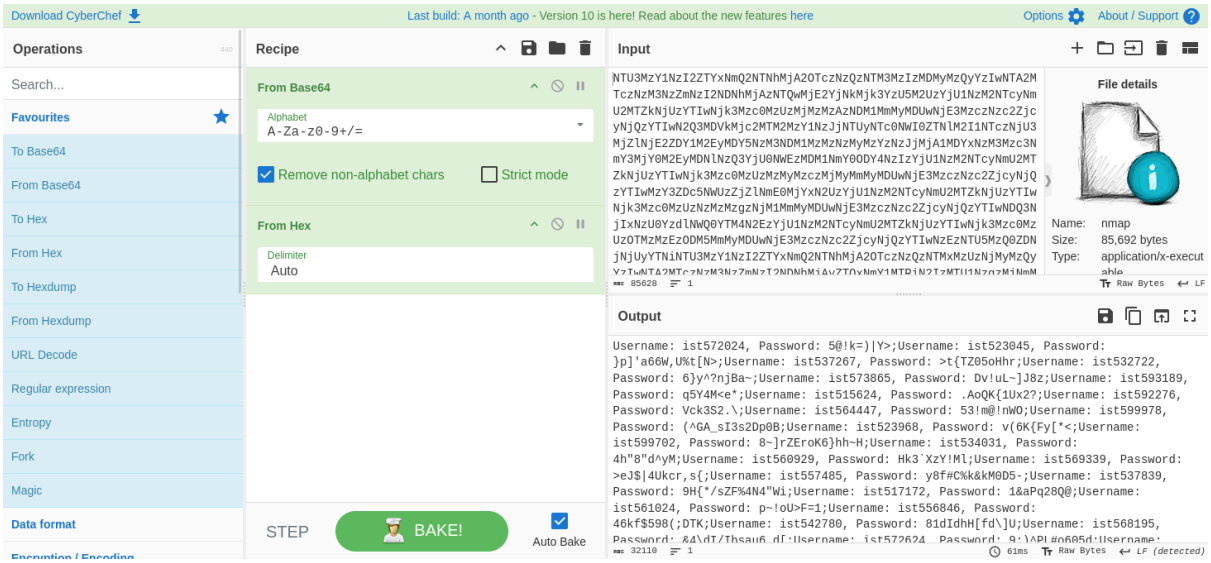


Figure 13 - Text decoding with CyberChef

secret 6 (rocket.png): tagus.png

Uploading the photo to https://fotoforensics.com, the photo appears to have something hidden in the upper left corner, in a triangle.

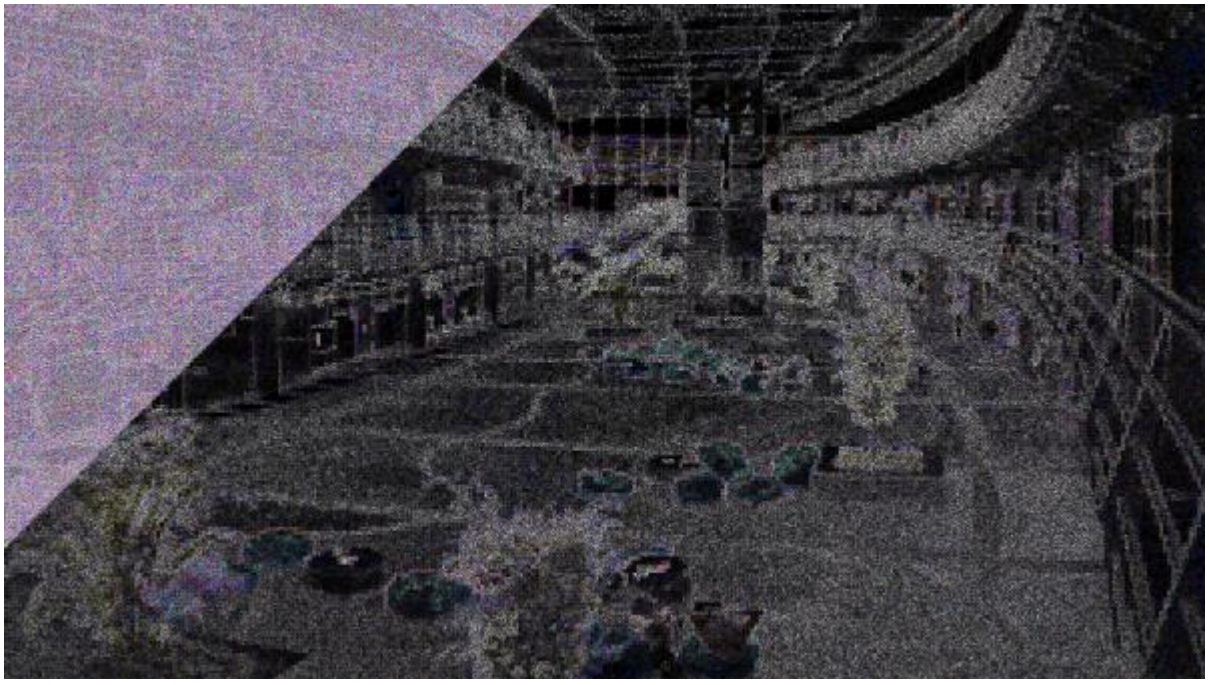


Figure 14 - photo forensics

After many iterations of a python script to read a number of LSB from each color channel, we found the desired hidden secret reading 5 LSB in each color channel, by reading the image diagonally from right to left, starting in the upper left corner, giving us a hidden image (rocket.png).

3 Analysis of relevant findings

3.1 Based on your analysis of the documents, did you find the stolen credentials? If so, describe how you identified them and provide details on the information you discovered.

We did find the stolen credentials in the nmap file, as described above in the **secret 5**, with a list of student ids and their respective passwords. They appeared to be hidden with a fake ELF file header and the credentials encoded in base64.

3.2 Did you uncover any additional concealed artifacts within the provided files? If so, explain how these artifacts were hidden and describe the methodology you used to extract them.

We found a hidden image explained in the **secret 6** section above, where it looks like a blueprint of some rocket ship, with an arrow pointing from the payload of the rocket ship to "ISTSat-1", Name of the ship is ARIANE6, manufactured in ARIANEGROUP and ESA agency.

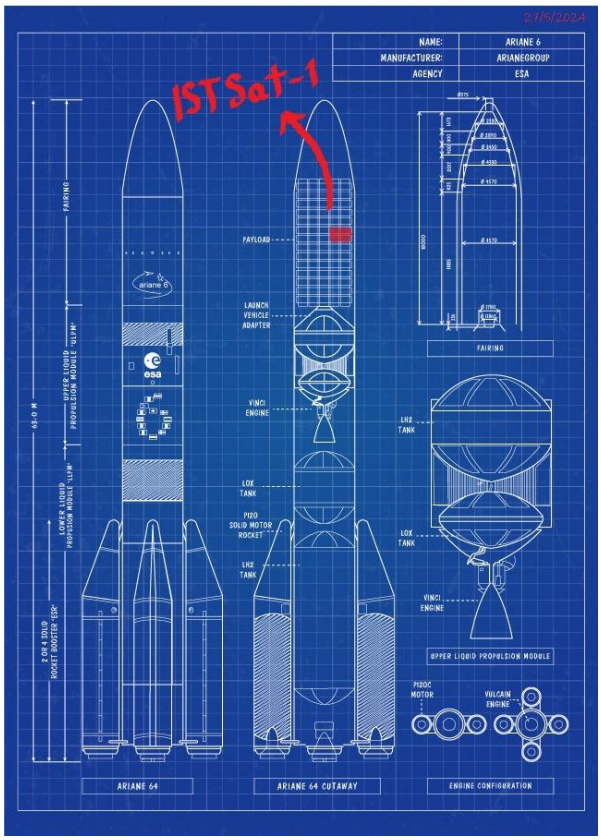
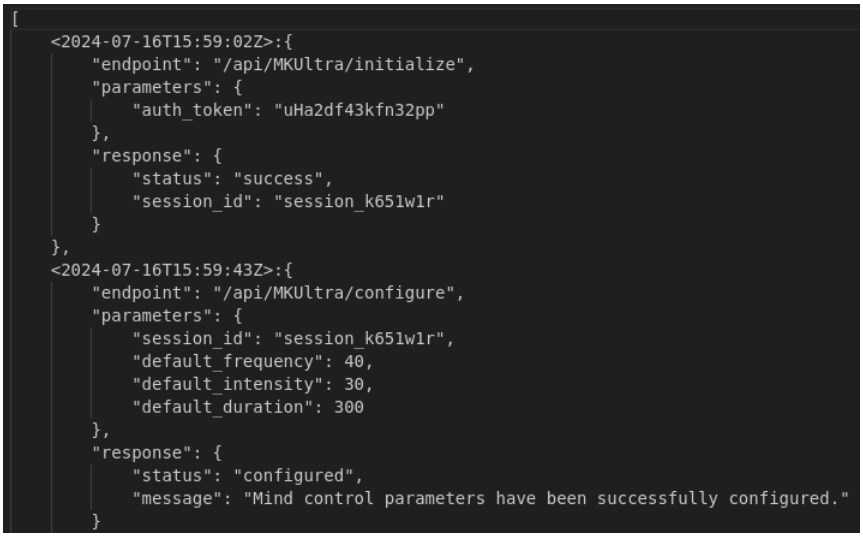


Figure 15 - rocket ship / satellite

The other concealed artifact is a json file (**secret 1 – Figure 16**) with api calls to different endpoints, like a login, a configuration of MKUltra with some parameters like default_frequency and default_duration, and */idea* endpoint also.

Other secret was a PDF file shown in **Figure 7** (**secret 3 and 4**) shows the sudden growth of Oeiras restaurants revenue and costumers.

Blank file (**secret 3 and 4**) is the documentation of MKUltra Mindo Control Component API.



```
[
    <2024-07-16T15:59:02Z>:{
        "endpoint": "/api/MKUltra/initialize",
        "parameters": {
            "auth_token": "uHa2df43kfn32pp"
        },
        "response": {
            "status": "success",
            "session_id": "session_k651w1r"
        }
    },
    <2024-07-16T15:59:43Z>:{
        "endpoint": "/api/MKUltra/configure",
        "parameters": {
            "session_id": "session_k651w1r",
            "default_frequency": 40,
            "default_intensity": 30,
            "default_duration": 300
        },
        "response": {
            "status": "configured",
            "message": "Mind control parameters have been successfully configured."
        }
```

Figure 16 - API calls secret

The file Best-intro.pdf (**secret 2 – Figure 4a**) is Virgolino's bank statement, with a statement date of 05/03/2024, covering the period from 01/01/2024 to 31/01/2024.

3.3 With a focus on the additional concealed secrets you recovered, analyze their content and relationships, and propose a possible interpretation of their meaning. Formulate a hypothesis regarding their significance and support it with the content of the recovered secrets. Additionally, prepare a timeline of the events as indicated by the recovered secrets.

Our hypothesis is that João Musk was chosen by a company with the goal to build that rocket ship (maybe satellite) – as found in **Figure 15** - and that company wants him to control the minds of students / workers that are working on that project, using MKULTRA Mind Control Component API (**Figure 11**).

To test if the mind control works, they tested first in random people in the area to go to a certain restaurant as shown in the **Figure 16**, API calls where it shows configuring the mind control device with some parameters and then a call targeting a region with an idea "I feel like going tonight to the restaurant..." and the response message "Idea implanted...".

The student credentials in **Figure 13** may be to select students for the mind control, accessing data like their address with those credentials.

3.4 Based on your findings, what recommendations would you make for the next steps in the investigation? Advise Mr. Ricardo Prado on the best course of action moving forward.

Based on our findings, we recommend the following steps to continue the investigation:

- understanding the reason for the theft, since the stolen credentials have been found, it's critical to identify the motive behind this crime. It's possible that João Musk was working alone, or he could be part of a larger criminal scheme.

- investigate who is Virgolino Gonçalves.

- investigate the possible illegal transfer of funds, as we suspect there was a suspicious financial transaction from ERCE.LTA to Virgolino Gonçalves on 05/01/2024.

- upon obtaining further confirmation of João's or another person's involvement, it is advisable to request a warrant to search their residence for additional evidence.