<table>
<tr><th colspan="2">CR7</th></tr>
<tr><td>AUTHOR
Mark Nudalo</td><td>OSINT, Password Cracking, Forensics, Python, Encoding/Decoding</td></tr>
<tr><td>800 – 1000pts</td><td>For MakeUC 2024</td></tr>
<tr><td colspan="2">Inside is a target location, unlock the file, investigate, and find the target location.

Read the “message.txt” it might help you.</td></tr>
</table>

Challenge Description:

The Player is given a password protected zip file along with a text file called “message .txt” or message.docx. The objective is to open this this zip and find the target. It contains broken images and a text file that is obscured as a different file called asm (although it’s only a text file). The message.txt is for story purposes only. Although, it may contain some hints that could help with solving the problem.

---

message.txt content:

Mission Brief: Intercepted Intel

Agent, we’ve intercepted a suspicious encrypted zip file from a hostile operative as he was attempting to flee from his London flat. We know for certain that he was planning on going outside of United Kingdom to one of the countries in the European Union. We believe that he was assigned a target location that could do a lot of damage. We must find that exact location at all costs!!!

The only clue left is this zip file. According to this hostile operative, the file is password protected. However, he is refusing to give the password. Instead, he keeps shouting “SUUIIIIIII” like a rockstar on some serious drugs. Our field agents described this man as “technologically challenged”. As such, we believe that we can crack the password with ease.

Your mission: crack the protection on this file, recover the intel, and report back with the exact target location. Time is ticking, and lives are at risk—failure is not an option.

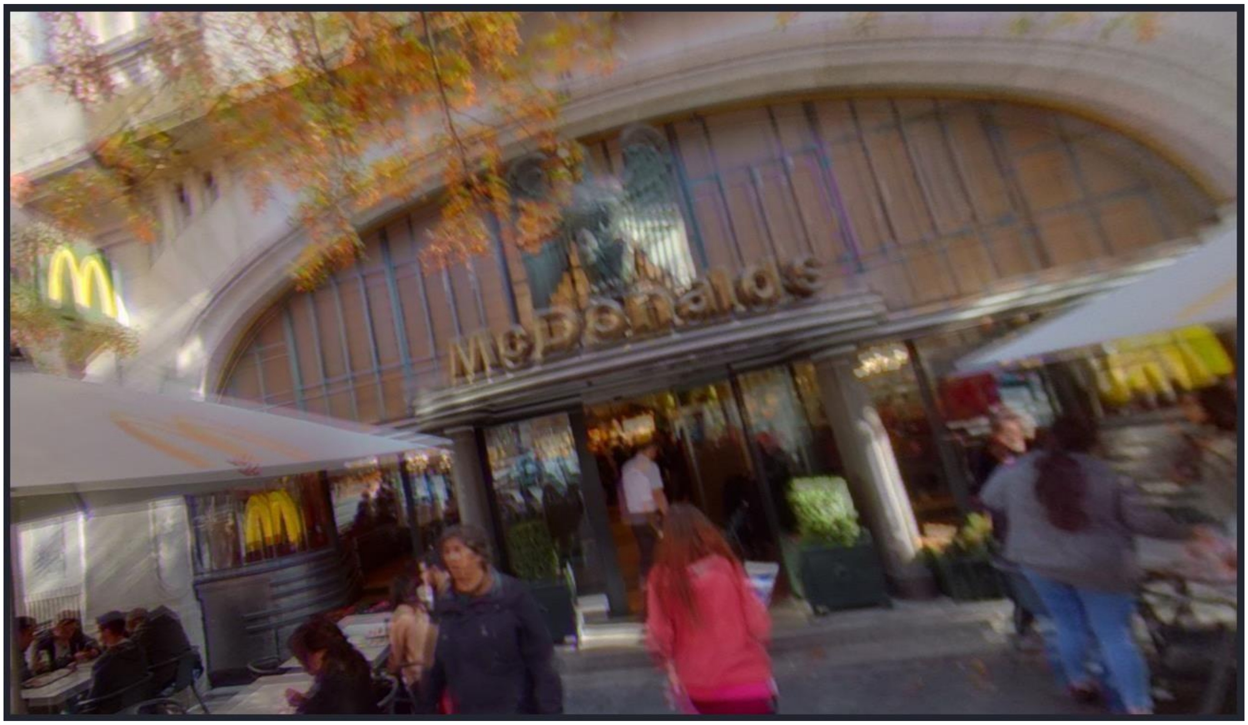
Agent McDay

Challenge Write Up

The target is the Sao Bento Station or Porto São Bento

Flag is: MAKEUC{Sao Bento Station} or MAKEUC{Porto São Bento} McDonald's Imperial in Porto, Portugal

The image below is the completed image problem. McDonald's Imperial in Porto, Portugal is known for the eagle statue in front lobby. However, this is not the target and is mainly use to find the real target (Sao Bento Station) which is nearby.



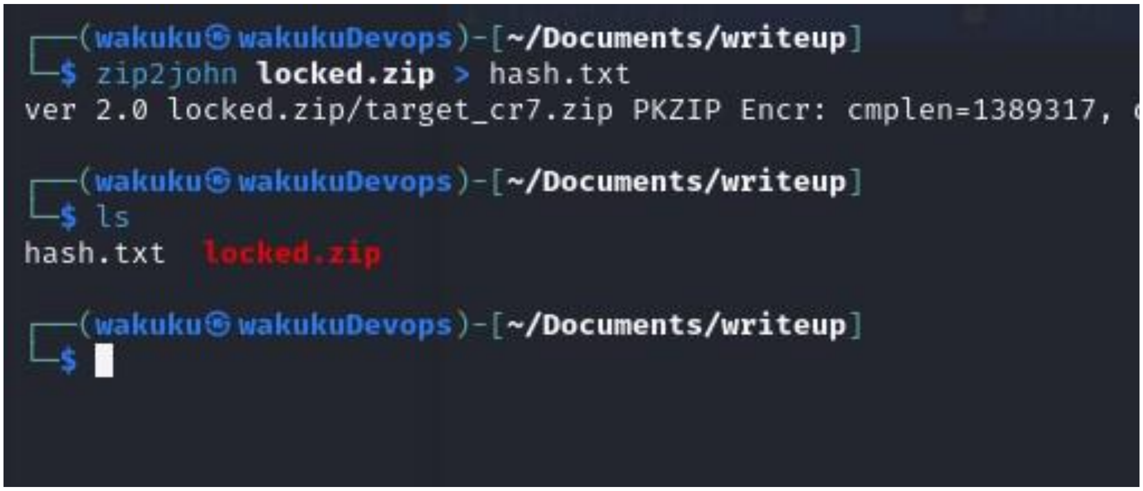
To solve this problem, follow these steps…

Figuring out the password

Install the John the Ripper and rockyou.txt wordlist

1. Using John the Ripper, we need to convert the zip file to a hash format.

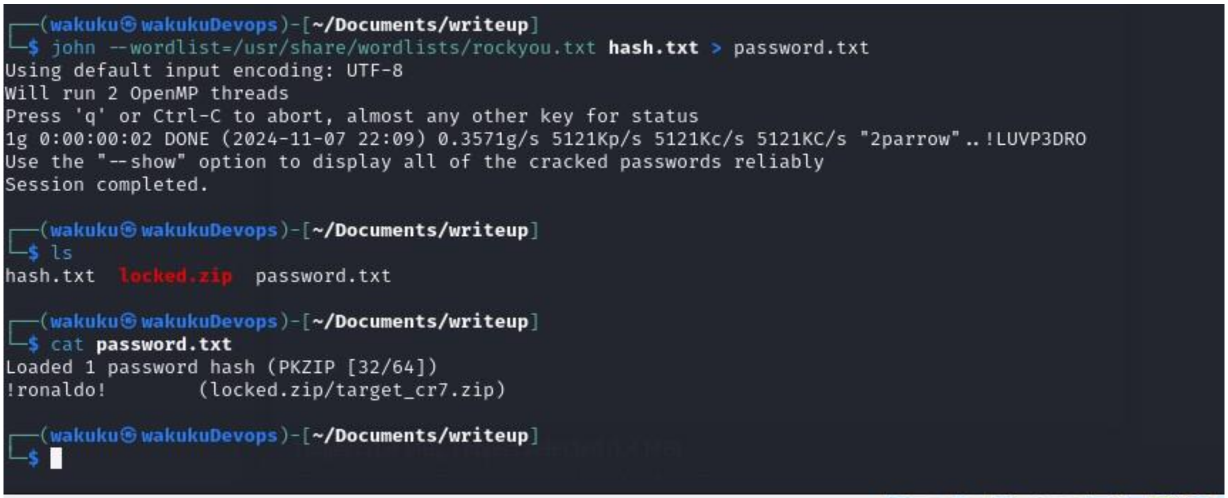
   $ zip2john locked.zip > hash.txt



```
┌──(wakuku㉿wakukuDevops)-[~/Documents/writeup]
└─$ zip2john locked.zip > hash.txt
ver 2.0 locked.zip/target_cr7.zip PKZIP Encr: cmplen=1389317, 

┌──(wakuku㉿wakukuDevops)-[~/Documents/writeup]
└─$ ls
hash.txt  locked.zip

┌──(wakuku㉿wakukuDevops)-[~/Documents/writeup]
└─$ 
```

2. We will use John the Ripper again with the rockyou.txt wordlist to get the password

   $ john --worldlist=/path/to/rockyou.txt hash.txt > password.txt



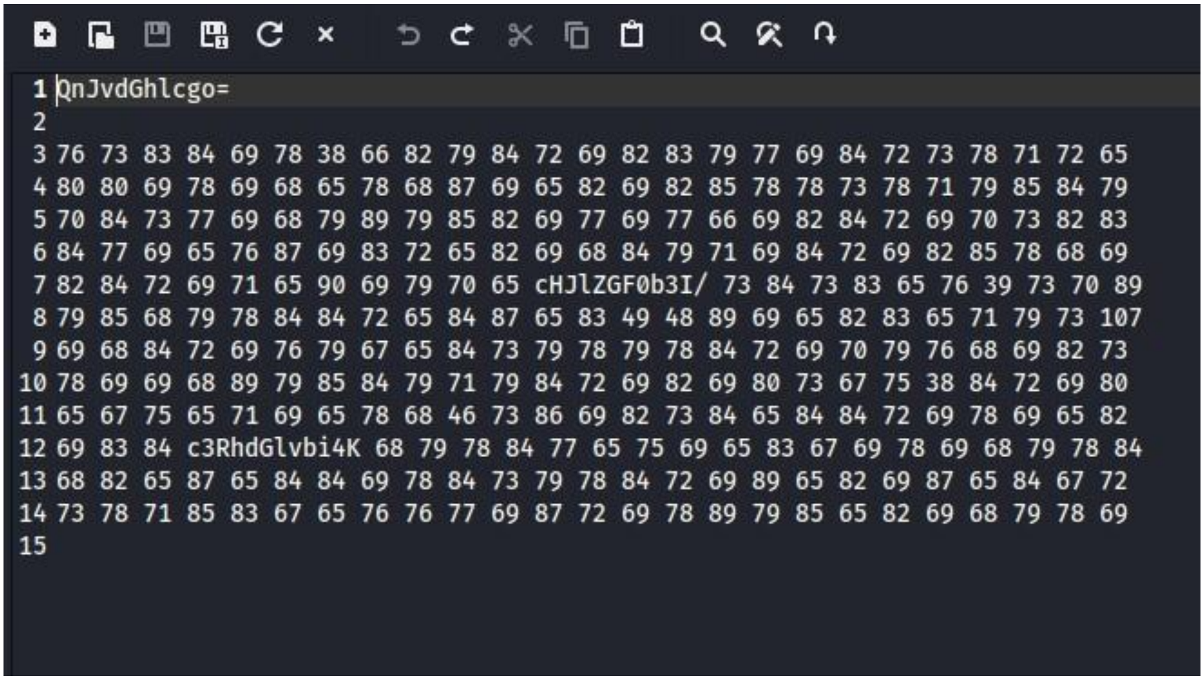
```
┌──(wakuku㉿wakukuDevops)-[~/Documents/writeup]
└─$ john --wordlist=/usr/share/wordlists/rockyou.txt hash.txt > password.txt
Using default input encoding: UTF-8
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
1g 0:00:00:02 DONE (2024-11-07 22:09) 0.3571g/s 5121Kp/s 5121Kc/s 5121KC/s "2parrow"..!LUVP3DRO
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

┌──(wakuku㉿wakukuDevops)-[~/Documents/writeup]
└─$ ls
hash.txt  locked.zip  password.txt

┌──(wakuku㉿wakukuDevops)-[~/Documents/writeup]
└─$ cat password.txt
Loaded 1 password hash (PKZIP [32/64])
!ronaldo!        (locked.zip/target_cr7.zip)

┌──(wakuku㉿wakukuDevops)-[~/Documents/writeup]
└─$ 
```

3. Extract the zip with the password [!ronaldo!] and you should see target_cr7.zip

4. P.S. I think you can find the password using hashcat but I don't have that command right now.

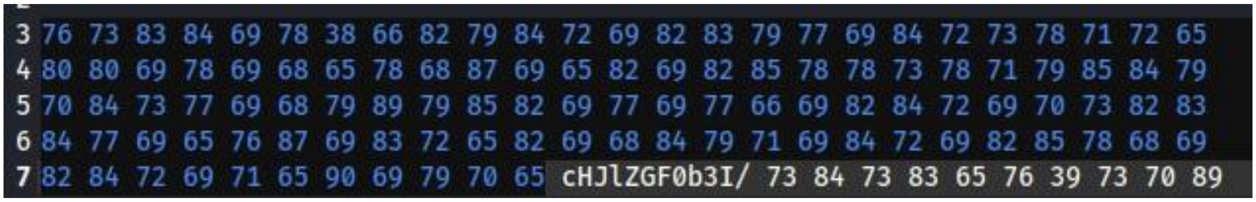
file_asm file

1. file_asm contains a cryptographic puzzle. First, it tries to fool the player that this file might be .asm extension. However, it is simply a text file you can open. This is the content



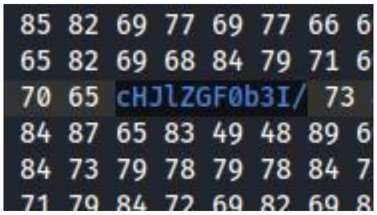
```
 1 QnJvdGhlcgo=
 2
 3 76 73 83 84 69 78 38 66 82 79 84 72 69 82 83 79 77 69 84 72 73 78 71 72 65
 4 80 80 69 78 69 68 65 78 68 87 69 65 82 69 82 85 78 78 73 78 71 79 85 84 79
 5 70 84 73 77 69 68 79 89 79 85 82 69 77 69 77 66 69 82 84 72 69 70 73 82 83
 6 84 77 69 65 76 87 69 83 72 65 82 69 68 84 79 71 69 84 72 69 82 85 78 68 69
 7 82 84 72 69 71 65 90 69 79 70 65 cHJlZGF0b3I/ 73 84 73 83 65 76 39 73 70 89
 8 79 85 68 79 78 84 84 72 65 84 87 65 83 49 48 89 69 65 82 83 65 71 79 73 107
 9 69 68 84 72 69 76 79 67 65 84 73 79 78 79 78 84 72 69 70 79 76 68 69 82 73
10 78 69 69 68 89 79 85 84 79 71 79 84 72 69 82 69 80 73 67 75 38 84 72 69 80
11 65 67 75 65 71 69 65 78 68 46 73 86 69 82 73 84 65 84 84 72 69 78 69 65 82
12 69 83 84 c3RhdGlvbi4K 68 79 78 84 77 65 75 69 65 83 67 69 78 69 68 79 78 84
13 68 82 65 87 65 84 84 69 78 84 73 79 78 84 72 69 89 65 82 69 87 65 84 67 72
14 73 78 71 85 83 67 65 76 76 77 69 87 72 69 78 89 79 85 65 82 69 68 79 78 69
15
```

2. This one [QnJvdGhlcgo=] can be decoded using Base 64. Decoded its “Brother” 3. This can be decoded using Javascript keycode OR as ASCII.



```
3 76 73 83 84 69 78 38 66 82 79 84 72 69 82 83 79 77 69 84 72 73 78 71 72 65
4 80 80 69 78 69 68 65 78 68 87 69 65 82 69 82 85 78 78 73 78 71 79 85 84 79
5 70 84 73 77 69 68 79 89 79 85 82 69 77 69 77 66 69 82 84 72 69 70 73 82 83
6 84 77 69 65 76 87 69 83 72 65 82 69 68 84 79 71 69 84 72 69 82 85 78 68 69
7 82 84 72 69 71 65 90 69 79 70 65 cHJlZGF0b3I/ 73 84 73 83 65 76 39 73 70 89
```

4. This can be decoded using Base 64



```
85 82 69 77 69 77 66 6
65 82 69 68 84 79 71 6
70 65 cHJlZGF0b3I/ 73
84 87 65 83 49 48 89 6
84 73 79 78 79 78 84 7
71 79 84 72 69 82 69 8
```

5. This is the decode text. The hint is marked yellow. Something about a a predator and a station. Decoding this will help pinpoint the right target

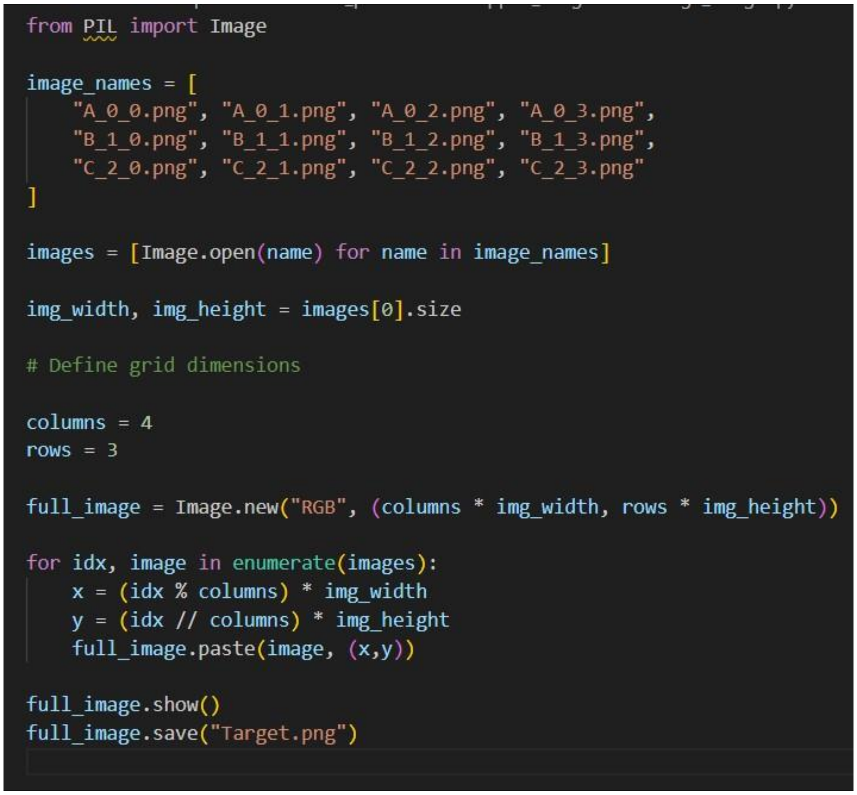
Listen up! Brother, something happened, and we are running out of time. Do you remember the first meal we shared together under the gaze of a predator It is alright if you don't, that was 10 years ago. I added the location on the folder. I need you to go there, pick up the package and deliver it at the nearest station.

Don't make a scene. Don't draw attention. They are watching us.

Call me when you are done.

Connecting the images together

1. The file contains 12 images with distinct file names. To solve this, we need to code in python and use the Python Imaging Library or PIL to connect the images as a whole.

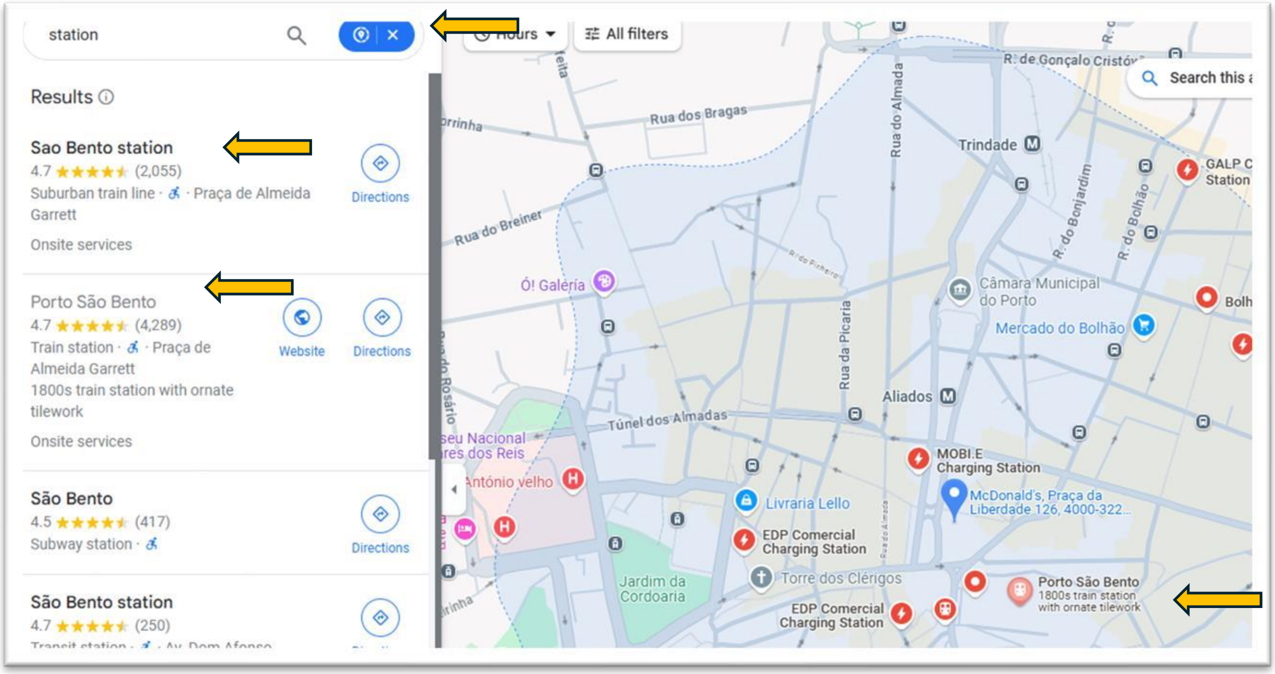


```python
from PIL import Image

image_names = [
    "A_0_0.png", "A_0_1.png", "A_0_2.png", "A_0_3.png",
    "B_1_0.png", "B_1_1.png", "B_1_2.png", "B_1_3.png",
    "C_2_0.png", "C_2_1.png", "C_2_2.png", "C_2_3.png"
]

images = [Image.open(name) for name in image_names]

img_width, img_height = images[0].size

# Define grid dimensions

columns = 4
rows = 3

full_image = Image.new("RGB", (columns * img_width, rows * img_height))

for idx, image in enumerate(images):
    x = (idx % columns) * img_width
    y = (idx // columns) * img_height
    full_image.paste(image, (x,y))

full_image.show()
full_image.save("Target.png")
```

2. Target.png should contain the whole image.

Figuring out the Target

1. Now with the whole image, we can use google images with google lens to figure out where this picture was taken. Although, you could also search for “mcdonalds with eagle” to find the location. However, as mentioned earlier, this is not the real target. Using google maps, we can use the “nearby” feature and search for “station”. The result should show Sao Bento Station.



station

Results ⓘ

Sao Bento station
4.7 ★★★★★ (2,055)
Suburban train line · ♿ · Praça de Almeida Garrett
Onsite services
Directions

Porto São Bento
4.7 ★★★★★ (4,289)
Train station · ♿ · Praça de Almeida Garrett
1800s train station with ornate tilework
Onsite services
Website
Directions

São Bento
4.5 ★★★★★ (417)
Subway station · ♿
Directions

São Bento station
4.7 ★★★★★ (250)