

# Writeup Heviosso nou gué N4t10n

Nous disposons d'un lien vers MEGA où l'on doit télécharger une vidéo

## Etape 1 :

Après téléchargement et visionnage de la vidéo on remarque une image avec une série de 0 et 1; c'est du binaire. Après décodage on obtient un lien youtube inversé. La mise en ordre donne ceci: <https://youtu.be/bDvam1tqAK8>

## Etape 2 :

- Le titre de la vidéo est encodé en **base32**. Le décodage nous donne **Do you see me?**

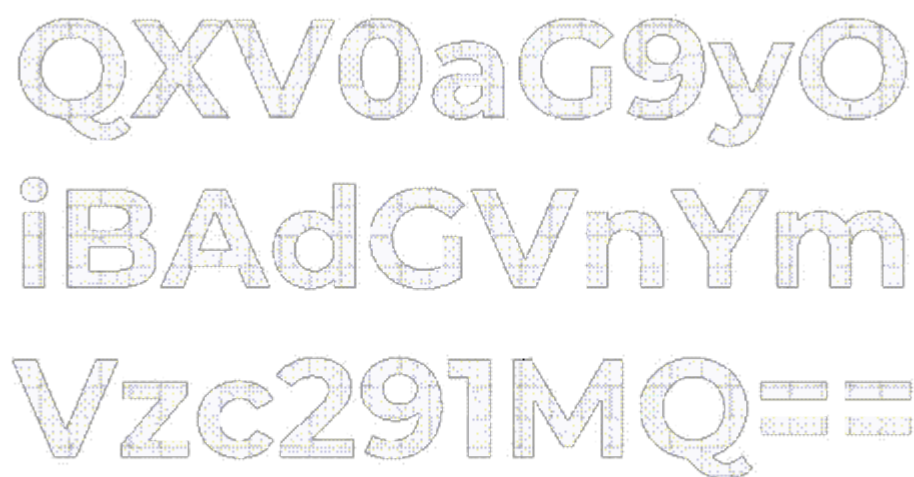
- Après téléchargement et visionnage de la vidéo on remarque qu'il y a une image blanche en fond et que le chaque partie est affichée étape par étape tout au long de la vidéo. L'idée m'est donc venu de convertir la vidéo en **GIF** et d'extraire chacune des frames de ce dernier. Il s'agira ensuite de superposer chacune de ces frames et tenant compte de l'opacité pour retrouver le l'image en fond

- La conversion en GIF s'est faite à l'aide de ce site: <https://image.online-convert.com/fr/convertir/mp4-en-gif> et on obtient le fichier **videoplayback.gif**

- L'extraction des frames s'est faite à l'aide de ce site: <https://ezgif.com/split> et on obtient le fichier zipé **frames.zip**

- La superposition des frames s'est faite à l'aide du script **heviosso.py**

- Le résultat final est l'image **combined\_image.png**



QXV0aG9yO  
iBAdGVnYm  
Vzc291MQ==

C'est du **base64**. Le décodage donne **Author: @tegbessou1**

### Etape 3 :

- Après un OSINT sur **@tegbessou1** on tombe sur le repo github suivant: <https://github.com/tegbessou1/oracle>
- Je fouille le dépôt et je tombe sur un ancien **commit** contenant un fichier **confidential.txt** et je le télécharge
- Le contenu du fichier est au format **hexdump** donc on le décode avec la commande **xxd**. La commande **file** nous permet de voir qu'il s'agit d'un fichier **wave** donc on change l'extension avec la commande **mv**

```
(samuel@kali)-[~/Documents/HACKERLAB_2023]
$ xxd -r confidential.txt > confidential

(samuel@kali)-[~/Documents/HACKERLAB_2023]
$ file confidential
confidential: RIFF (little-endian) data, WAVE audio, Microsoft PCM, 16 bit, stereo 44100 Hz

(samuel@kali)-[~/Documents/HACKERLAB_2023]
$ mv confidential confidential.wav
```

- Le son ne donne rien de concret. Mais à l'aide de l'outil **stegolsb** on obtient ceci

```
(samuel@kali)-[~/Documents/HACKERLAB_2023]
$ stegolsb wavsteg -r -i confidential.wav -o data.txt -n 1 -b 10000
Files read in 0.00s
Recovered 10000 bytes in 0.00s
Written output file in 0.01s

(samuel@kali)-[~/Documents/HACKERLAB_2023]
$ cat data.txt
Find my e-mail address and send me a message with the TIC-TAC-TOE challenge answer in the subject line.
,++<+8~+UU2L+2$+f1+f!++233Dd+6I+L++2fbd+f+f#233+f`+3339+l+2fS1+c0+DF!+f"8+1+++9++?+*****+
,++0+3+s+8++X++B+-
```

### Etape 4 :

Il faut donc retrouver l'adresse mail de **tegbessou1** et lui envoyer un message en mettant en objet la réponse du challenge **TIC-TAC-TOE**

- La réponse du challenge **TIC-TAC-TOE** est:  
**CTF\_50uRC3m4P\_J5\_072147408013208**

- Pour retrouver le mail de l'utilisateur, on se sert de l'api de github en allant sur le lien suivant: <https://api.github.com/users/tegbessou1/events/public>. La recherche du champ **email** nous permet de ressortir l'adresse mail suivante: [th3t0ul41960@gmail.com](mailto:th3t0ul41960@gmail.com)

- Dès l'envoi du mail avec les paramètres requis, on reçoit cette réponse:  
**PGS\_T4eq13af\_Q3F\_7erf0ef\_743285253** (Il faut faire un **CTRL+A** pour le voir puisque la couleur est blanche)

- Le message est en **ROT13**. Le décodage nous donne le flag:

**CTF\_G4rd13ns\_D3S\_7res0rs\_743285253**