

> TRICKY COMMUNICATIONS

BY: ES

DIFFICULTY: EASY

CATEGORY: STEGO

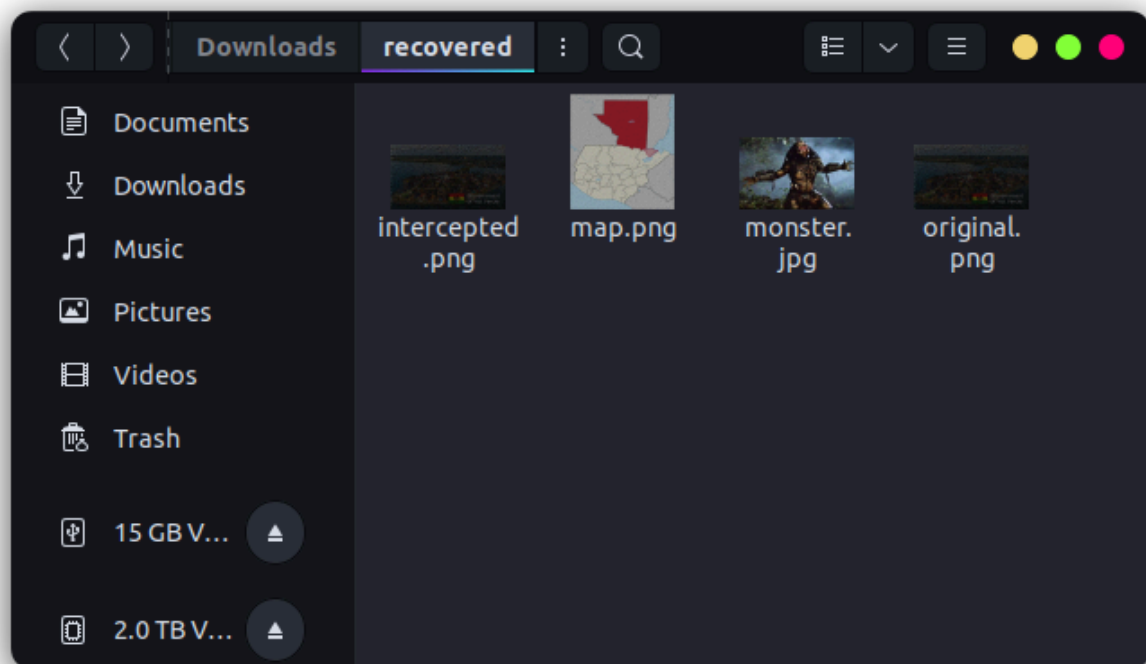
SOLVES: 0 (SORRY)

Description: Some images were intercepted in communication between potential bad actors. Figure out what they were talking about. Flag format is a btc address.

Artifact: recovered.zip

SOLUTION:

Decompress the archive and we see 4 images:



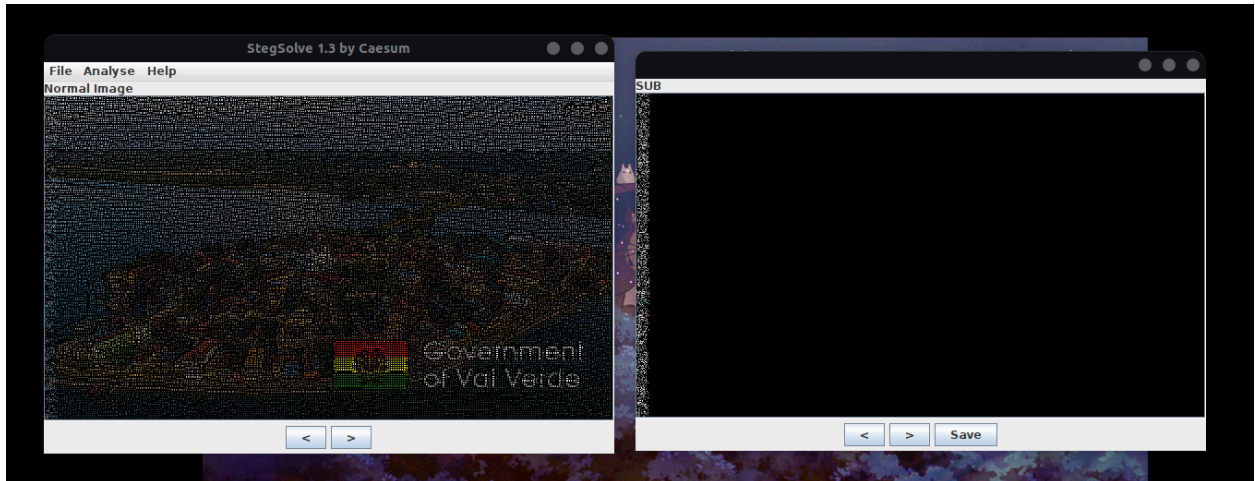
What I would typically do is try and run tools like foremost, steghide, binwalk, and other stego tools on the images in the archive, and on not finding anything delve a little bit deeper.

You may notice that in this archive there are two images: `intercepted.png` and `original.png` that are the same image but have different noise on the left.



A method of hiding data is in the difference between two images. This is the thread you are supposed to pull on, the other two images in the archive are red herrings.

If we use stegsolve with the Image Combiner tool we can get the difference (SUB) between the 2 images:



Shown more closely:



We can see that the noise along the size of the image is binary data.

We can write a script to extract it, converting black pixels to 0s and white pixels to 1s:

```
from PIL import Image

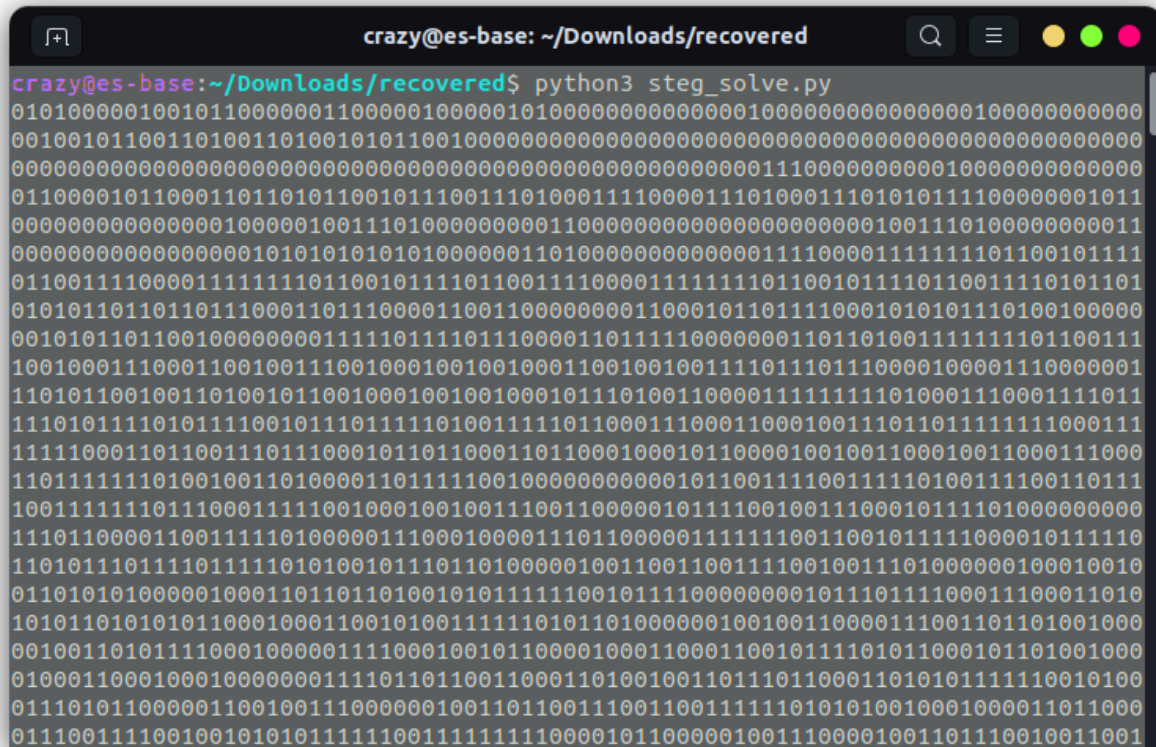
img_orig = Image.open("solved.bmp")
rgb_im = img_orig.convert('RGB')

width = 615
height = 350

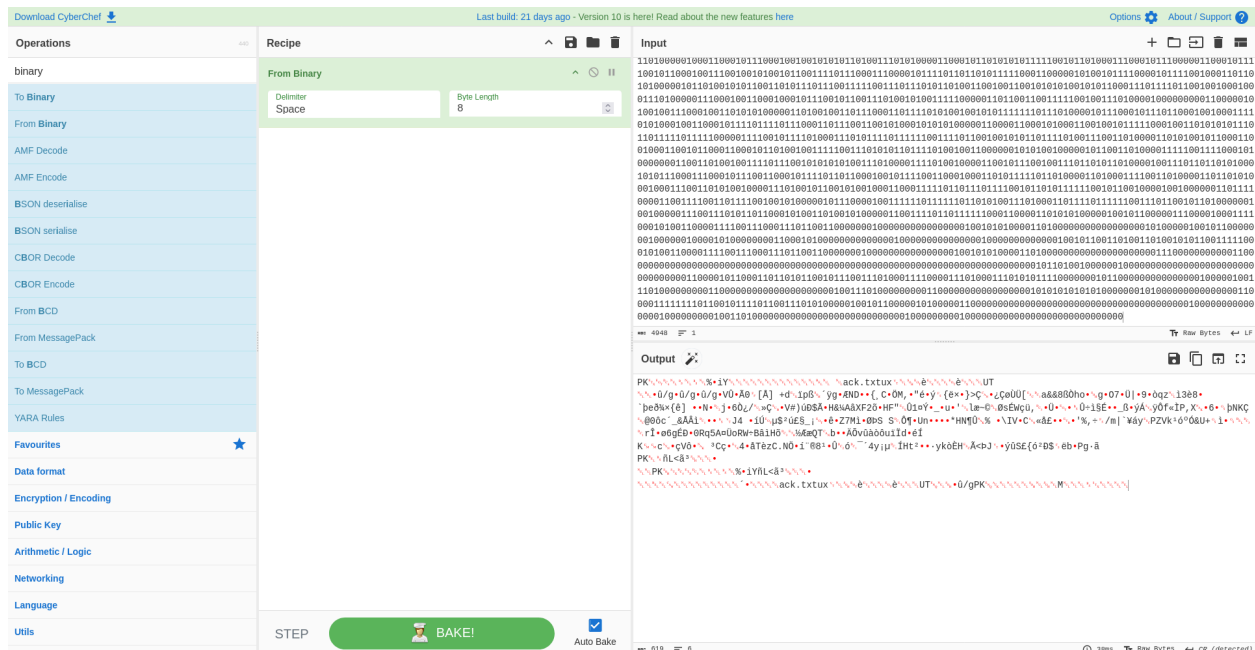
pix = img_orig.load()
output = ""
for j in range(width):
    for i in range(height):
        bin = pix[j,i]
        if str(bin) == "(0, 0, 0)":
            output += "0"
        else:
            output += "1"

print(output)
```

We get the following:



Lets drop it into good olde cyberchef:

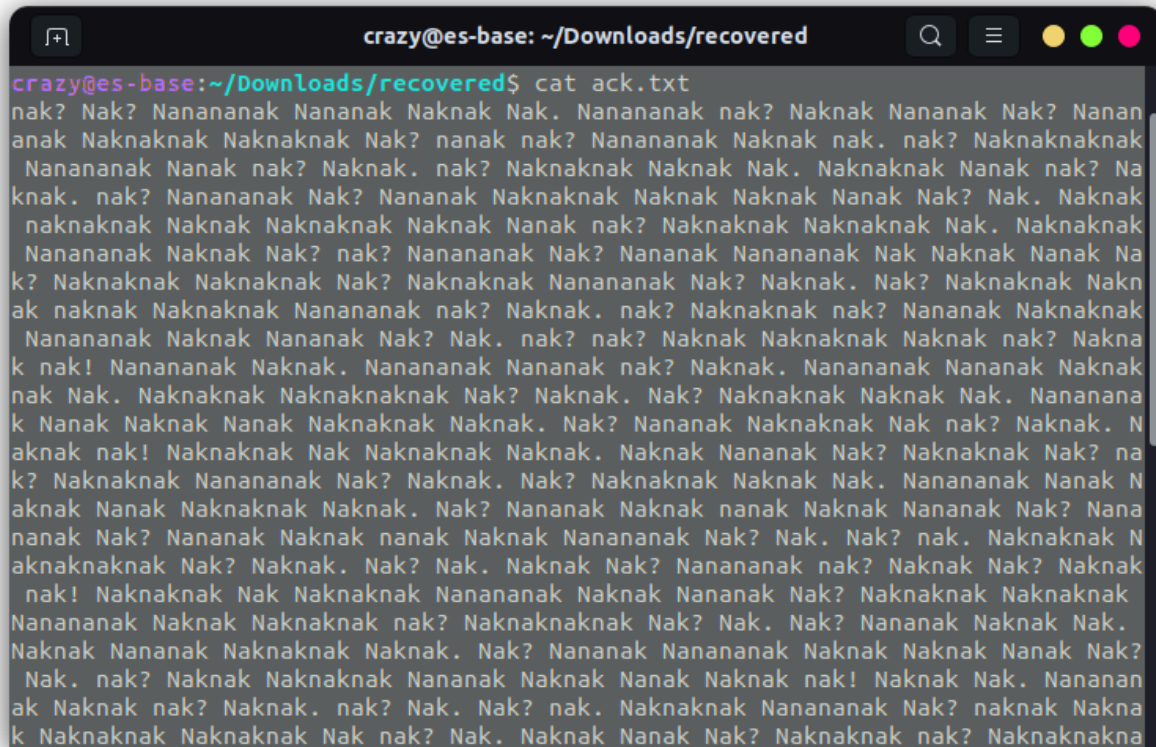


We can see that this is a zip file, but when we try to open it after downloading it, it will be corrupted. We must have the correct number of trailing null-bytes in order to get a zip file. This can be easily done with a script, adding 0's after the last 1 in the output of the previous script (steg_solve.py) until you get a valid zip file.

The total size of the zip file should be **4952** bytes.

Extract the contents of the archive.

When we extract the contents of the zip file we get a text file ack.txt:

A terminal window titled 'crazy@es-base: ~/Downloads/recovered' displays the contents of a file named 'ack.txt'. The text is a repeating pattern of 'Nan' and 'nak' characters, such as 'Nan? Nan? Nanananak Nananak Naknak Nak. Nanananak nak? Naknak Nananak Nak? Nanan', which is a form of steganography. The terminal window has a dark background and standard macOS window controls (red, yellow, green buttons) in the top right corner.

```
crazy@es-base:~/Downloads/recovered$ cat ack.txt
nak? Nak? Nanananak Nananak Naknak Nak. Nanananak nak? Naknak Nananak Nak? Nanan
anak Naknaknak Naknaknak Nak? nanak nak? Nanananak Naknak nak. nak? Naknaknaknak
Nanananak Nanak nak? Naknak. nak? Naknaknak Naknak Nak. Naknaknak Nanak nak? Na
knak. nak? Nanananak Nak? Nananak Naknaknak Naknak Naknak Nanak Nak? Nak. Naknak
naknaknak Naknak Naknaknak Naknak Nanak nak? Naknaknak Naknaknak Nak. Naknaknak
Nanananak Naknak Nak? nak? Nanananak Nak? Nananak Nanananak Nak Naknak Nanak Na
k? Naknaknak Naknaknak Nak? Naknaknak Nanananak Nak? Naknak. Nak? Naknaknak Nakn
ak naknak Naknaknak Nanananak nak? Naknak. nak? Naknaknak nak? Nananak Naknaknak
Nanananak Naknak Nananak Nak? Nak. nak? nak? Naknak Naknaknak Naknak nak? Nakna
k nak! Nanananak Naknak. Nanananak Nananak nak? Naknak. Nanananak Nananak Naknak
nak Nak. Naknaknak Naknaknaknak Nak? Naknak. Nak? Naknaknak Naknak Nak. Nananan
k Nanak Naknak Nanak Naknaknak Naknak. Nak? Nananak Naknaknak Nak nak? Naknak. N
aknak nak! Naknaknak Nak Naknaknak Naknak. Naknak Nananak Nak? Naknaknak Nak? na
k? Naknaknak Nanananak Nak? Naknak. Nak? Naknaknak Naknak Nak. Nanananak Nanak N
aknak Nanak Naknaknak Naknak. Nak? Nananak Naknak nanak Naknak Nananak Nak? Nana
nanak Nak? Nananak Naknak nanak Naknak Nanananak Nak? Nak. Nak? nak. Naknaknak N
aknaknaknak Nak? Naknak. Nak? Nak. Naknak Nak? Nanananak nak? Naknak Nak? Naknak
nak! Naknaknak Nak Naknaknak Nanananak Naknak Nananak Nak? Naknaknak Naknaknak
Nanananak Naknak Naknaknak nak? Naknaknaknak Nak? Nak. Nak? Nananak Naknak Nak.
Naknak Nananak Naknaknak Naknak. Nak? Nananak Nanananak Naknak Naknak Nanak Nak?
Nak. nak? Naknak Naknaknak Nananak Naknak Nanak Naknak nak! Naknak Nak. Nananan
ak Naknak nak? Naknak. nak? Nak. Nak? nak. Naknaknak Nanananak Nak? naknak Nakna
k Naknaknak Naknaknak Nak nak? Nak. Naknak Nanak Nak? Naknaknak nak? Naknaknakna
```

We visit my favorite site: [dcode.fr](https://www.dcode.fr)

And use their cipher identifier to figure out what in the tarnation this is:

<https://www.dcode.fr/cipher-identifier>



CIPHER IDENTIFIER
Cryptography › Cipher Identifier

ENCRYPTED MESSAGE IDENTIFIER

★ CIPHERTEXT TO RECOGNIZE ⓘ
naknak Naknak Naknak. Naknak Nanak Naknak nak! Nak? Naknak
Naknaknak Naknaknak Naknak Nak? Naknak nak. nak? Naknak
Nanananak Naknak Nak? naknak Naknak Naknaknak Naknaknak Nak
Nak? nak! Naknak Nanak Nak? Naknaknak Nanananak nak?
Naknaknak Naknaknaknak Naknak Nanak Nak? Nanak

★ CLUES/KEYWORDS (IF ANY)

▶ ANALYZE

See also: [Frequency Analysis — Index of Coincidence](#)

SYMBOLS IDENTIFIER

▶ Go to: [Symbols Cipher List](#)

Answers to Questions (FAQ)

What is a cipher identifier? (Definition)
A encryption detector is a computer tool designed to recognize encryption/
encoding from a text message. The detector performs cryptanalysis.

Summary

- ★ Encrypted Message Identifier
- ★ What is a cipher identifier? (Definition)
- ★ How to decrypt a cipher text?
- ★ How to recognize a cipher?
- ★ Why does the detector display a warning?
- ★ Why does the analyzer/recognizer not detect my cipher method?
- ★ How does the cipher identifier work?

Similar pages

- ★ Index of Coincidence
- ★ Frequency Analysis

Search for a tool

★ SEARCH A TOOL ON DCODE BY KEYWORDS:
e.g. type 'random'

★ BROWSE THE [FULL DCODE TOOLS' LIST](#)

Results

dCode's analyzer suggests to investigate:

1	1
Nak Nak (Duckspeak)	■
Substitution Cipher	■
Shift Cipher	□
Homophonic Cipher	□
Ook!	□

Cipher Identifier - dCode
Tag(s) : Cryptography, Cryptanalysis, dCode

Its Duckspeak, quack



Use the Nak Nak (Duckspeak) tool to decode it:



NAK NAK (DUCKSPEAK)
Cryptography › Substitution Cipher › Nak Nak (Duckspeak)

NAK NAK DECODER

★ NAK NAK CIPHERTEXT ⓘ
nak? Nak? Nanananak Nananak Naknak Nak. Nanananak nak?
Naknak Nananak Nak? Nanananak Naknaknak Naknaknak Nak?
nanak nak? Nanananak Naknak nak. nak? Naknaknaknak
Nanananak Nanak nak? Naknak. nak? Naknaknak Naknak Nak.
Naknaknak Nanak nak? Naknak. nak? Nanananak Nak? Nananak

★ RESULTS FORMAT ⓘ **STRING OF PRINTABLE CHARACTERS (ASCII/UNICODE)**

☐ HEXADECIMAL 00-7F-FF

☐ FILE TO DOWNLOAD

▶ DECRYPT

See also: [ASCII Code](#)

NAK NAK ENCODER

★ NAK NAK PLAINTEXT ⓘ
T2h5bCwKSnZ1YWhqYSBvaHogaWxsdsB0aGtsIGlsYWrsb
HUgem92Y2xzIGH1ayBpYmpybGEsIGH1ayBkbCBkcHNzIH
d5dmpsbGsgZHBhbyB6aHVramh6YXNsLgpXaGZ0bHVhIGR
wc3MgaWwgemx1YSBhdia4Nm9PRWFQsXhWaDhtem1tYnJ1
RUd1cHpQMGVZcVp4bnpScKhkaHBhIGlieWFvbHkgcHV6Y
XliamFwdnV6LgpMaG5zbA

Nak Nak (Duckspeak) - dCode
Tag(s) : Substitution Cipher

Share

Use cyberchef to finish decoding and we can see the BTC address:

Download CyberChef [Download CyberChef](#) Last build: 21 days ago - Version 10 is here! Read about the new features here Options About / Support

Operations	Recipe	Input
rot		TZh5bCwKSnZ1YwhqYSBvaHogawksdSB8a0t'sIG1sYwRsbHUGem92Y2xzIGh1ay8pYmpyb6EsIGh1ay8KbcBkcHNzIHd5dmpsb6sgZHBhbyB6aHVraah8YXN5LgpXaGZ2bGVhIGRwc3RgaWwgemx1YSBhd1A4Nm9KwFQsXhwaDhtem1YnJ1Rud1chpQMGVZcVp4bnpSCkhkaHBHIG11eWFvbkGcHV6YX11amFwdmV0LgpKaG52b4
ROT13	From Base64 Alphabet A-Za-z0-9+/= <input checked="" type="checkbox"/> Remove non-alphabet chars	
ROT47		
ROT8000	<input type="checkbox"/> Strict mode	
Rotate left	ROT13	
Rotate image	<input checked="" type="checkbox"/> Rotate lower case chars <input checked="" type="checkbox"/> Rotate upper case chars	
Rotate right	<input type="checkbox"/> Rotate numbers Amount 19	
ROT13 Brute Force		
ROT47 Brute Force		
Parse ObjectID timestamp		
Avro to JSON		
From UNIX Timestamp		
From Octal		
Protobuf Decode		
Protobuf Encode		
Drop bytes		
From Float		
Remove Diacritics		
Remove null bytes		
Remove whitespace		
From HTML Entity		
From Hex Content		

STEP Auto Bake

246 1 245-246 (1 selected) Raw Bytes LF

Output

Here,
Contact has been made between shovel and bucket, and we will proceed with sandcastle.
Payment will be sent to 86hHXtIBqOa8fsffuknXZnisI0xRjSqgsK
Await further instructions.
Eagle

184 5 Raw Bytes LF

Flag: 86hHXtIBqOa8fsffuknXZnisI0xRjSqgsK

DONE. THATS A WRAP :)



FEEL FREE TO DM ME WITH FEEDBACK ON THIS CHALLENGE :)