

Category:

Steganography

Name:

Steg2.png

Message:

You are provided with an image named “steg2.png”. The flag for this challenge is written in this image. However, the image dimensions were altered by mistake! Your mission is to capture the flag by diving into the binary depths and restoring “steg2.png”!

Hint:

- Tried poking around with LSB decoding yet? It might reveal more than you expect! Psst... the original height of “steg2.png” could be your clue!
- Ever wondered what makes a PNG tick? Dive into the PNG file structure and pinpoint which bytes hold the height information. Once you tweak that, you're one step closer.
- CRC errors raining on your parade? Recalculate the CRC and give “steg2.png” a second life. pngcheck might just be a good company!

Objective:

Your task is to restore the image dimension by editing the binary file of “steg2.png” to read the flag. This requires understandings of LSB manipulation and PNG image structure.

Instructions:

1. Start by loading “steg2.png” in your preferred environment. You will see the yellow image of a person carrying a flag.



2. Look at the meta data of "steg2.png" with tools such as ExifTool, Strings, pngcheck etc. to see the basic file information. For example, using ExifTool, you will find the image size as 636 (width) and 555 (height) pixels.

```
$ exiftool steg2.png
```

```
Title           : steg2.png
Metadata Date   : 
Copyright      : AJCCBC_CTF2024
Image Size      : 636x555
Megapixels      : 0.353
```

3. Like the previous challenge with "steg1.png," the image hides a string using the LSB technique. In this case, the least significant (0) bits are modified in the Red and Green channels, as hinted by the yellow color (the combination of Red and Green light) the original image.
4. In StegOnline, one of the most popular tools, upload the image and then choose Extract Files/Data. In the window, select the least significant bit (0 bit) in Red and Green channels. The result should show the string **"The original height is 580 pixel"**. This suggests that the current height 555 pixel should be restored to 580 pixels to capture the flag.

StegOnline

[Upload](#) [Image Home](#) [CTF Checklist](#) [About](#)

[Back to Home](#)

Extract Data

Here you can extract data hidden inside of the image. Select some bits and adjust the settings appropriately. The final extracted data is checked against some basic file headers, and so the filetype can be automatically determined.

Please note that Alpha options are only available if the image contains transparency.

	R	G	B
7	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
6	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
5	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
1	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Pixel Order

Row ▾

Bit Order

LSB ▾

Bit Plane Order

R ▾ G ▾ B ▾

Trim Trailing Bits

No ▾

Go

Results

No file types identified.

The results below only show the first 2500 bytes. Select "Download" to obtain the full data.

Ascii (readable only):

```
The original height is 580 pixel .....
```

5. Although there are several ways to modify the image size, let's edit the binary data in this scenario as the challenge message requests. Speaking about the PNG file structure, it is made up of several chunks to form an image as shown below.

PNG File

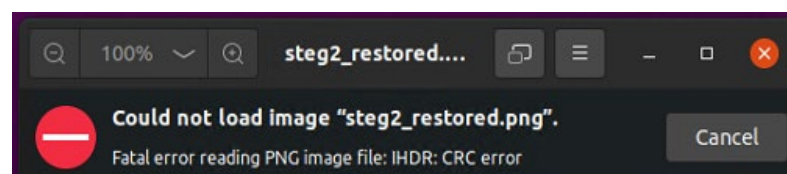
Signature	Fixed sequence of bytes for all PNG files ("89 50 4E 47 0D 0A 1A 0A")
IHDR	The Header chunk defining the critical image information like width, height etc.
IDAT	Image Data chunk containing the actual compressed image data
IEND	Image End chunk with the mark of the end of the PNG file

The IHDR chunk contains important file information such as width, height, bit depth etc. In another word, you can modify the image height by properly editing this chunk.

- Open preferred Hex editor and load "steg2.png". Following to PNG file signature ("89 50 4E 47 0D 0A 1A 0A"), the IHDR chunk starts. IHDR chunk starts from length (13 bytes represented as "00 00 00 0D") and then the chunk type code for IHDR ("49 48 44 52"). The next 4 bytes ("00 00 02 7C") represent the image width (636 pixel), while another 4 bytes ("00 00 02 2B") represent the image height (555 pixel). Note that 636 and 555 are interpreted to 27C and 22B in hex respectively. To resize the height from 555 to 580 pixel, modify the 4 bytes from "00 00 02 2B" to "00 00 02 44" as 580 is interpreted to 244 in hex. Use any Hex/Binary editor to modify the binary such as Ghex, HxD and so forth.

Signature	<code>chen@ubuntu:~/Downloads\$ xxd steg2.png</code>
IHDR	<pre> 00000000: 8950 4e47 0d0a 1a0a 0000 000d 4948 4452 .PNG.....IHDR 00000010: 0000 027c 0000 022b 0806 0000 0075 6644 +....uFD 00000020: 1900 0000 0174 4558 7454 6974 6c65 0073 tEXtTitle.s 00000030: 7465 6732 2e70 6e67 286f 11df 0000 0018 teg2.png(o..... 00000040: 7445 5874 436f 7079 7269 6768 7400 414a tEXtCopyright.AJ 00000050: 4343 4243 5f43 5446 3230 3234 8353 5212 CCBC_CTF2024.SR. 00000060: 0000 517e 4944 4154 789c eddd 7f70 d5e5 ..Q~IDATx....p.. 00000070: 9dff fd57 7e40 021c 2460 80a8 5181 c48a ...W~@.\$.`..Q... 00000080: 556b 5851 b18b 2e6d a9d0 f9b6 b338 4ba7 UkXQ...m....8K. 00000090: 50ed 573a dbba 8ddb de63 67db d2ce d669 P.W:....cg....i 000000a0: bd75 c742 bbb7 eddd 56bc 5bbf 537a d796 .u.B....V.[.Sz.. 000000b0: ec94 9d75 ee82 75bb 6c49 1515 d758 u..u.lI...X 000000c0: b162 4930 68c4 0011 0244 12e0 9073 ff71 .bI0h....D...s.q 000000d0: f524 2727 2739 d727 e773 cef5 f9f1 7ccc .\$('9.'s.... . </pre>
IDAT	

- In some cases, the modified PNG file is recognized as corrupted. This error is detected by the checksum function CRC stored in IHDR chunk which detects the lack of data integrity. It is caused because the height related binary data in IHDR chunk was modified.



You need to recalculate the CRC value before opening the modified image. Using pngcheck, a

tool to verify the integrity of PNG files, you can see the current CRC value is “8cc4f996”. As the predefined CRC value is “75664419”, modify it into “8cc4f996” using Hex/Binary editor.

```
$ pngcheck <modified_steg2.png>
```

```
chen@ubuntu:~/Downloads$ pngcheck steg2_edited.png
steg2_edited.png CRC error in chunk IHDR (computed 8cc4f996, expected 75664419)
ERROR: steg2_edited.png
```

	Signature
	chen@ubuntu:~/Downloads\$ xxd steg2_edited.png
IHDR	00000000: 8950 4e47 0d0a 1a0a 0000 000d 4948 4452 .PNG.....IHDR
	00000010: 0000 027c 0000 0244 0806 0000 0075 6644D.....ufD
	00000020: 1930 0000 0f74 4558 7454 6974 6c65 0073tEXtTitle.s
	00000030: 7465 6732 2e70 6e67 286f 11df 0000 0018 teg2.png(o.....
	00000040: 7445 5874 436f 7079 7269 6768 7400 414a tEXtCopyright.AJ

	Signature
	chen@ubuntu:~/Downloads\$ xxd steg2_edited_CRC.png
IHDR	00000000: 8950 4e47 0d0a 1a0a 0000 000d 4948 4452 .PNG.....IHDR
	00000010: 0000 027c 0000 0244 0806 0000 008c c4f9D.....
	00000020: 9630 0000 0f74 4558 7454 6974 6c65 0073tEXtTitle.s
	00000030: 7465 6732 2e70 6e67 286f 11df 0000 0018 teg2.png(o.....
	00000040: 7445 5874 436f 7079 7269 6768 7400 414a tEXtCopyright.AJ

8. Now you should be able to see the original image with flag written at the bottom.



Flag is:

CSG_FLAG{Broaden_your_horizons_to_find_the_FLAG!35}