

Forensics CTF 2

Platform: picoCTF 2025

Challenge Name: RED

Category: Forensics

Difficulty: Easy

Submitted By: Gurleen Kaur Brar

Objective

The goal of the challenge was to extract hidden data embedded within an image file and decode it to reveal the CTF flag. This task involved detecting steganographic content, identifying encodings, and using the right tools to decode it.

RED

Easy

Forensics

picoCTF 2025

browser_webshell_solvable

AUTHOR: SHUAILIN PAN (LECONJUROR)

Hints ?

Description

1 2 3

RED, RED, RED, RED

Download the image: [red.png](#)

Files and Tools Used

- **File Provided:** `red.png` (image file)
- **Tools Used:**
 - Kali Linux Terminal
 - `zsteg` – for steganographic data extraction
 - CyberChef – for Base64 decoding

Step-by-Step Process

Step 1: Run **zsteg** on the Image

The image likely contained hidden data using steganography. To analyze it, I ran:

```
zsteg red.png
```

This command listed several embedded data entries in different color channels and bits. One of them included Base64-encoded strings.

```
(gurleen@kali) - [~/Downloads]
$ zsteg red.png

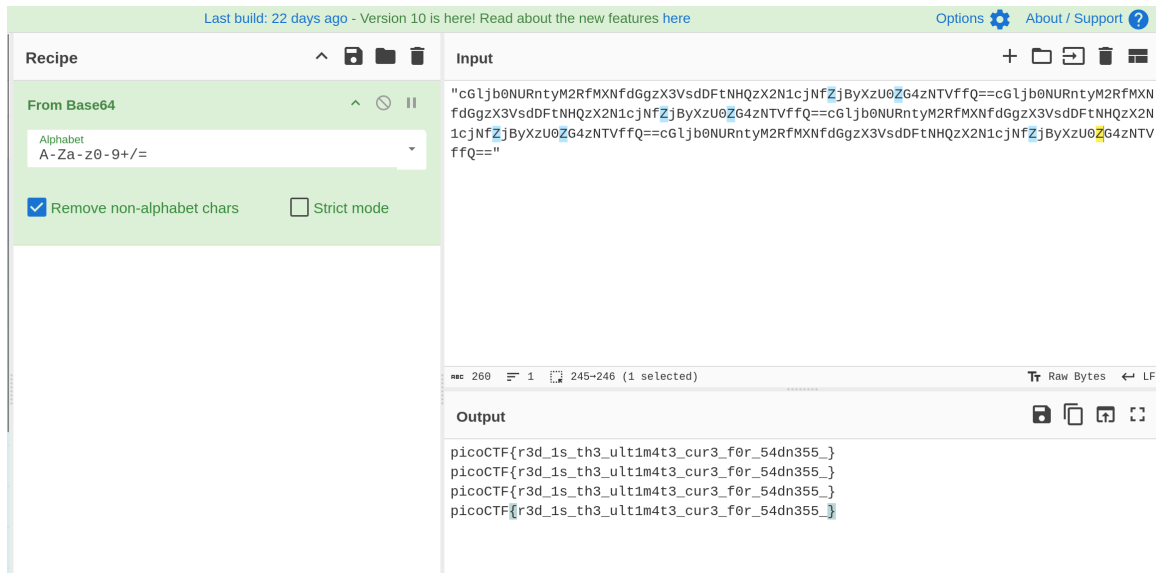
meta.Poem .. text: "Crimson heart, vibrant and bold,\nHearts flutter at your sight,\nEvenings glow softly red,\nCherries burst with sweet life.\nKisses linger with your warmth.\nLove deep as meadow,\nScarlet leaves falling softly,\nBold in every stroke."
b1,rgba,lsb,xy .. text: "c01jb0NURatyM2RfMmNfdGgzX3VsdDFlNHQzX2N1c jNFZjByX2U0ZG4zNTVffQ==c01jb0NURatyM2RfMmNfdGgzX3VsdDFlNHQzX2N1c jNFZjByX2U0ZG4zNTVffQ==c01jb0NURatyM2RfMmNfdGgzX3VsdDFlNHQzX2N1c jNFZjByX2U0ZG4zNTVffQ==c01jb0NURatyM2RfMmNfdGgzX3VsdDFlNHQzX2N1c jNFZjByX2U0ZG4zNTVffQ=="
b1,rgba,msb,xy .. file: OpenPGP Public Key
b2,g,lsb,xy .. text: "ETqUETPETUUTqTUUTDgPD00000RE"
b2,rgb,lsb,xy .. file: OpenPGP Secret Key
b2,bgr,msb,xy .. file: OpenPGP Public Key
b2,rgba,lsb,xy .. file: OpenPGP Secret Key
b2,rgba,msb,xy .. text: "Cikiiii"
b2,abgr,lsb,xy .. file: OpenPGP Secret Key
b2,abgr,msb,xy .. text: "iiiaakikk"
b3,rgba,msb,xy .. text: "#wb#wp#7p"
b3,abgr,msb,xy .. text: "7r'wb#7p"
b4,b,lsb,xy .. file: 0421 Alliant compact executable not stripped
```

Step 2: Copy and Decode Base64 String

I copied one of the longer Base64 strings. To decode it, I used **CyberChef** with the "From Base64" recipe.

Step 3: Get the Flag

The output of the decoded Base64 string in CyberChef revealed the flag:



Flag Submitted

The extracted flag was:

```
picoCTF{r3d_1s_th3_ult1m4t3_cur3_f0r_54dn355_}
```

The flag was successfully submitted and confirmed as correct.

