



Challenge Name: Hidden Whispers — 150pts (Steganography)

The screenshot shows a challenge card for "Hidden Whispers" worth 150 points. It includes a description about a journalist transmitting a "normal" file before disappearing, and a quote: "There's something beneath the surface. Can you uncover the secret and extract the flag?". A note at the bottom says "Trust nothing. Inspect everything." There are buttons for "Data" (with a download icon) and "Flag".

🧙 Recon & Initial Inspection:

I downloaded the challenge file named Data using the following **wget** command:

```
(kali㉿kali)-[~/Desktop/cybersecure/stegnogarphy/Hidden_Whispers]
└─$ wget https://cybersecure-x.ctfd.io/files/d3b3ef03e3a950daeadfc028fd1bbe0/Data?token=eyJ1c2VyX2lkIjoyMTIsInRlYW1faWQ1ojEwNSwiZmlsZV9pZCI6MTR9.aEc3_w.zu3lUAzlKaOCKkJYzJ8WFwnMqhs --O Data
--2025-06-09 16:20:28-- https://cybersecure-x.ctfd.io/files/d3b3ef03e3a950daeadfc028fd1bbe0/Data?token=eyJ1c2VyX2lkIjoyMTIsInRlYW1faWQ1ojEwNSwiZmlsZV9pZCI6MTR9.aEc3_w.zu3lUAzlKaOCKkJYzJ8WFwnMqhs
Resolving cybersecure-x.ctfd.io (cybersecure-x.ctfd.io) ... 45.55.122.20, 138.197.58.222
Connecting to cybersecure-x.ctfd.io (cybersecure-x.ctfd.io) [45.55.122.20]:443 ... connected.
HTTP request sent, awaiting response ... 302 FOUND
Location: https://29832a894184429789dc1139b5658a4e.s3.amazonaws.com/d3b3ef03e3a950daeadfc028fd1bbe0/Data?response-content-disposition=attachment%3B%20filename%3DData&response-cache-control=max-age%3D36000X-Amz-Algorithm=AWS4-HMAC-SHA256X-Amz-Credential=AKIAQYMCZ24V83JEZERx2F0250609%2Fus-east-1%2F3%2Faws4_request&X-Amz-Date=20250609T200000Z&X-Amz-Expires=36000X-Amz-SignedHeaders=host&X-Amz-Signature=B42Fc2c6d99b29849c2ade39fdaed1c015d764eab7379c86b6aF24de002153 [following]
--2025-06-09 16:20:29-- https://29832a894184429789dc1139b5658a4e.s3.amazonaws.com/d3b3ef03e3a950daeadfc028fd1bbe0/Data?response-content-disposition=attachment%3B%20filename%3DData&response-cache-control=max-age%3D36000X-Amz-Algorithm=AWS4-HMAC-SHA256X-Amz-Credential=AKIAQYMCZ24V83JEZERx2F0250609%2Fus-east-1%2F3%2Faws4_request&X-Amz-Date=20250609T200000Z&X-Amz-Expires=36000X-Amz-SignedHeaders=host&X-Amz-Signature=B42Fc2c6d99b29849c2ade39fdaed1c015d764eab7379c86b6aF24de002153
Resolving 29832a894184429789dc1139b5658a4e.s3.amazonaws.com (29832a894184429789dc1139b5658a4e.s3.amazonaws.com) ... 3.5.27.165, 3.5.8.150, 16.15.216.0, ...
Connecting to 29832a894184429789dc1139b5658a4e.s3.amazonaws.com (29832a894184429789dc1139b5658a4e.s3.amazonaws.com) [3.5.27.165]:443 ... connected.
HTTP request sent, awaiting response ... 200 OK
Length: 896 [binary/octet-stream]
Saving to: 'Data'

Data                                         [ <=>                                     ]   896 --.-KB/s    in 0s
2025-06-09 16:20:30 (45.5 MB/s) - 'Data' saved [896/896]
```

Next, I checked the file type using the **file** command:

```
(kali㉿kali)-[~/Desktop/cybersecure/stegnogarphy/Hidden_Whispers]
└─$ file Data
Data: data
```

So, it wasn't recognized as any known file type.

Checked for strings inside:

```
(kali㉿kali)-[~/Desktop/cybersecure/stegnogarphy/Hidden_Whispers]
$ strings Data
W%KW
XXXX
GIDATx
ZW4! L
!gLsP
Y{uRk
qu      8
}ru5
>;?Hy
IEND bbb
        _banner.pn...  capture.pc...  Data
```

Found PNG chunks (GIDAT, IEND), but the header started with XXXX, which is wrong.

Used hex dump to confirm:

```
(kali㉿kali)-[~/Desktop/cybersecure/stegnogarphy/Hidden_Whispers]
$ xxd Data | head -5
00000000: 5725 4b57 0d0a 1a0a 0000 000d 5858 5858 W%KW.....XXXX
00000010: 0000 0258 0000 0064 0802 0000 00d5 23d9 ... X ... d.....#.
00000020: 7400 0003 4749 4441 5478 9ced dadd 729b t ... GIDATx....r.
00000030: 3014 4651 d3e9 fbbf 32bd f00c 4311 3afc 0.FQ....2 ... C...
00000040: 1863 27df 5a57 3421 204c a31d 610f e338 .c'.ZW4! L...a..8
```

The PNG signature bytes should be: **89 50 4E 47 0D 0A 1A 0A** — but here it's something else.

Fixing the Corrupted PNG Header:

Replaced the first 8 bytes with the correct PNG signature:

```
(kali㉿kali)-[~/Desktop/cybersecure/stegnogarphy/Hidden_Whispers]
$ printf '\x89\x50\x4E\x47\x0D\x0A\x1A\x0A' | dd of=Data bs=1 seek=0 count=8 conv=noTrunc
8+0 records in
8+0 records out
8 bytes copied, 0.000205646 s, 38.9 kB/s
```

Repaired the **IHDR** chunk name at offset 12:

```
(kali㉿kali)-[~/Desktop/cybersecure/stegnogarphy/Hidden_Whispers]
$ printf 'IHDR' | dd of=Data bs=1 seek=12 count=4 conv=noTrunc
4+0 records in
4+0 records out
4 bytes copied, 0.000262836 s, 15.2 kB/s
```

Copied the repaired file to a new name and checked the file type again:

```
(kali㉿kali)-[~/Desktop/cybersecure/stegnogarphy/Hidden_Whispers]
$ cp Data repaired.png && file repaired.png
repaired.png: PNG image data, 600 x 100, 8-bit/color RGB, non-interlaced
```

Opened the repaired PNG to confirm it displays normally:

The screenshot shows a terminal window on a Kali Linux desktop. The terminal output is as follows:

```
kali@kali: ~/Desktop/cybersecure/steganography/Hidden_Whispers$ xxhd Data | head -2
00000000: 5725 4b57 0d0a 1a0a 0000 000d 5858 5858 W%KW.....XXXX
00000010: 0000 0258 0000 0004 0802 0000 0005 23d9 ...X...d....#.
(kali㉿kali)-[~/Desktop/cybersecure/steganography/Hidden_Whispers]$ xxhd Data | head -5
00000000: 5725 4b57 0d0a 1a0a 0000 000d 5858 5858 W%KW.....XXXX
00000010: 0000 0258 0000 0004 0802 0000 0005 23d9 ...X...d....#.
00000020: 7400 0003 4744 4441 5478 9ced dadd 729b t...GIDATX....r.
00000030: 3014 4651 036e fbf1 32bd F00c 4311 3afc 0.FQ....2...C...
00000040: 1663 276f 5a57 3421 204c a31d 610f e338 .c.ZW41L...a.a.8
(kali㉿kali)-[~/Desktop/cybersecure/steganography/Hidden_Whispers]$ printf '\x89\x50\x4E\x47\x0D\x0A\x1A\x0A' | dd of=Data bs=1 seek=0 count=8 conv=notrunc
8+0 records in
8+0 records out
8 bytes copied, 0.000205646 s, 38.9 kB/s
(kali㉿kali)-[~/Desktop/cybersecure/steganography/Hidden_Whispers]$ printf 'IHDR' | dd of=Data bs=1 seek=12 count=4 conv=notrunc
4+0 records in
4+0 records out
4 bytes copied, 0.000262836 s, 15.2 kB/s
(kali㉿kali)-[~/Desktop/cybersecure/steganography/Hidden_Whispers]$ cp Data repaired.png
file repaired.png
repaired.png: PNG image data, 600 x 100, 8-bit/color RGB, non-interlaced
(kali㉿kali)-[~/Desktop/cybersecure/steganography/Hidden_Whispers]$ xdg-open repaired.png
(kali㉿kali)-[~/Desktop/cybersecure/steganography/Hidden_Whispers]$ 
(kali㉿kali)-[~/Desktop/cybersecure/steganography/Hidden_Whispers]$ 
(kali㉿kali)-[~/Desktop/cybersecure/steganography/Hidden_Whispers]$ eog repaired.png
^C
(kali㉿kali)-[~/Desktop/cybersecure/steganography/Hidden_Whispers]$ cp Data repaired.png & file repaired.png
repaired.png: PNG image data, 600 x 100, 8-bit/color RGB, non-interlaced
(kali㉿kali)-[~/Desktop/cybersecure/steganography/Hidden_Whispers]$ eog repaired.png

```

A separate window titled "repaired.png" shows a white image with a small black rectangle in the center containing the text "flag{broken_bytes_but_not_the_spirit}".

So, we got our flag.

FLAG: flag{broken_bytes_but_not_the_spirit}

✓ Why We Used These Commands

- `printf '\x89\x50\x4E\x47\x0D\x0A\x1A\x0A'` → This adds the **PNG file signature** at the beginning. Without it, the system doesn't even know it's an image.
- `printf 'IHDR'` → This adds the **IHDR chunk** at the right spot. It's like telling the image viewer, "Hey, this is how to read me!"

Both commands fix the broken parts of the image so we can open it and check if anything is hidden inside.

~By Team justahacker