Challenge | 26 Solves | ✕

Blindsighted

25

Stego

Once, my dad told me that the answers to all our questions lie within the night sky.

Unlock Hint for 5 points

⬇ night_sky.png

| Flag | Submit |

We are given a png image, shown below.

Looking at the top of image we can see what lloks like braille.

Looks pretty straight forward let's decode this braille string and see what we get.

I'm going to use an app on my iphone to help with this, Braille Go, it allows me to enter each braille character and decodes them for me. Below is the whole braille string with the decoded text.

⠊⠶⠆⠧ ⠓⠓⠆⠧ ⠓⠶⠆⠧ ⠑⠋⠧ ⠑⠓⠆⠧ ⠊⠑⠧ ⠋⠶⠆⠧ ⠛⠶⠆⠧ ⠚⠋⠧ ⠶⠶⠆⠧ ⠊⠑⠧ ⠋⠓⠆⠧ ⠙⠶⠆⠧ ⠑⠋⠧ ⠑⠶⠆⠧ ⠊⠑⠧
⠶⠊⠧ ⠶⠊⠧ ⠊⠶⠆⠧ ⠊⠑⠧ ⠶⠶⠆⠧ ⠑⠋⠧ ⠓⠶⠆⠧ ⠚⠋⠧ ⠊⠑⠧ ⠓⠶⠆⠧ ⠶⠊⠧ ⠑⠑⠧ ⠊⠑⠧ ⠛⠶⠆⠧ ⠚⠋⠧ ⠶⠶⠆⠧ ⠊⠑⠧
⠊⠶⠆⠧ ⠛⠊⠧ ⠓⠶⠆⠧ ⠑⠋⠧ ⠋⠓⠆⠧ ⠙⠶⠆⠧ ⠊⠑⠧ ⠶⠓⠆⠧ ⠑⠓⠆⠧ ⠚⠋⠧
⠊⠶⠆⠧ ⠉⠶⠆⠧ ⠓⠛⠧ ⠋⠚⠧ ⠛⠙⠧ ⠶⠶⠆⠧ ⠶⠶⠆⠧ ⠙⠶⠆⠧ ⠚⠓⠆⠧ ⠚⠓⠆⠧ ⠊⠓⠆⠧ ⠶⠛⠧

I??V HH?V H??V EFV EH?V IEV F??V G??V JFV ???V IEV FH?V D??V EFV E??V IEV ?IV ?IV I??V IEV ???V
EFV H??V JFV IEV H??V ?IV EEV IEV G??V JFV ???V IEV I??V GIV H??V EFV FH?V D??V IEV ?H?V EH?V JFV
I??V C??V HGV FJV GDV ???V ???V D??V JH?V JH?V IH?V ?GV

Now this mean nothing to me but the one thing that stands out is the v at the end of every group.
Looking up how braille works we can see the numbers in braille always start with ⠼ which is the

reverse of what we have, so let's flip the whole string and decode again. Below is the new braille string and decoded text.

⠼⠃⠃ ⠼⠁⠚⠑ ⠼⠁⠚⠓ ⠼⠁⠚⠓ ⠼⠁⠁⠋ ⠼⠁⠁⠁ ⠼⠁⠁⠃ ⠼⠋⠛ ⠼⠓⠙ ⠼⠛⠚ ⠼⠁⠃⠉ ⠼⠁⠁⠑ ⠼⠙⠓ ⠼⠁⠚⠊
⠼⠁⠚⠁ ⠼⠊⠑ ⠼⠁⠁⠋ ⠼⠁⠚⠙ ⠼⠙⠊⠁ ⠼⠁⠁⠚ ⠼⠑⠛ ⠼⠁⠁⠑ ⠼⠊⠑ ⠼⠁⠃⠁ ⠼⠙⠓ ⠼⠁⠁⠛ ⠼⠊⠑ ⠼⠊⠁
⠼⠑⠃ ⠼⠁⠁⠚ ⠼⠊⠑ ⠼⠙⠓ ⠼⠁⠁⠚ ⠼⠙⠊⠁ ⠼⠁⠃⠁ ⠼⠊⠑ ⠼⠁⠁⠑ ⠼⠑⠁ ⠼⠑⠁ ⠼⠊⠑ ⠼⠁⠁⠊ ⠼⠙⠊⠁ ⠼⠁⠁⠋
⠼⠁⠚⠙ ⠼⠊⠑ ⠼⠁⠃⠁ ⠼⠙⠓ ⠼⠁⠁⠛ ⠼⠁⠁⠙ ⠼⠊⠑ ⠼⠁⠚⠊ ⠼⠙⠊⠁ ⠼⠁⠁⠚ ⠼⠁⠚⠚ ⠼⠁⠃⠑

72 105 108 108 116 111 112 67 84 70 123 115 48 109 101 95 116 104 49 110 57 115 95 121 48 117 95 99 52 110 95 48 110 49 121 95 115 51 51 95 119 49 116 104 95 121 48 117 114 95 109 49 110 100 125

This is looking better. This looks the decimal values which we can then convert to ascii. To do this we can use a nice online tool call cyberchef https://gchq.github.io/CyberChef/ and pick from decimal.

Recipe 💾 📁 🗑 **Input** start: 194 / end: 194 / length: 0   length: 194 / lines: 1   ➕ 📁 ⤴ 🗑 ▬

From Decimal ⊘ ‖

Delimiter
Space

Support signed values

```
72 105 108 108 116 111 112 67 84 70 123 115 48 109 101 95 116 104
49 110 57 115 95 121 48 117 95 99 52 110 95 48 110 49 121 95 115 5
51 95 119 49 116 104 95 121 48 117 114 95 109 49 110 100 125
```

Output time: 5ms / length: 55 / lines: 1 💾 📋 📤 ↩ ⟦⟧

HilltopCTF{s0me_th1n9s_y0u_c4n_0n1y_s33_w1th_y0ur_m1nd}

FLAG: HilltopCTF{s0me_th1n9s_y0u_c4n_0n1y_s33_w1th_y0ur_m1nd}

You can also decode the decimal in kali using the below command

```
kali@kali:~/Pictures$ while read i; do printf \\$(printf "%o" $i);done < Braille
HilltopCTF{s0me_th1n9s_y0u_c4n_0n1y_s33_w1th_y0ur_m1nd}kali@kali:~/Pictures$ █
```