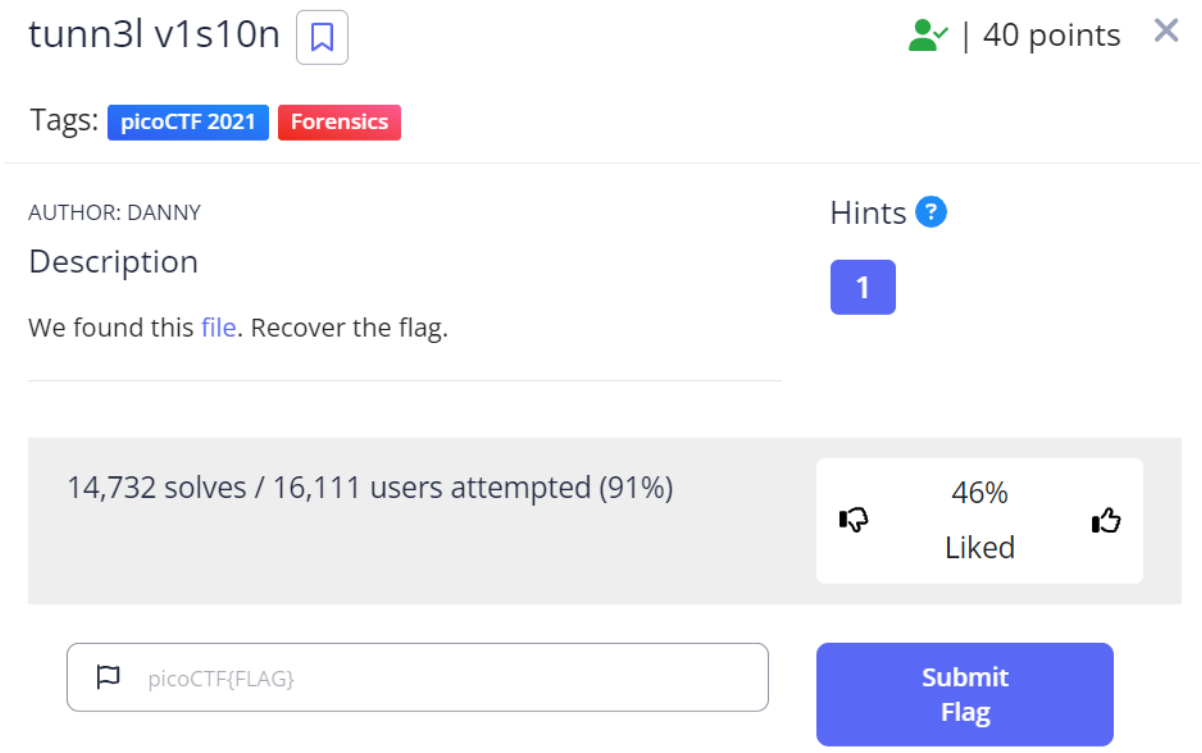
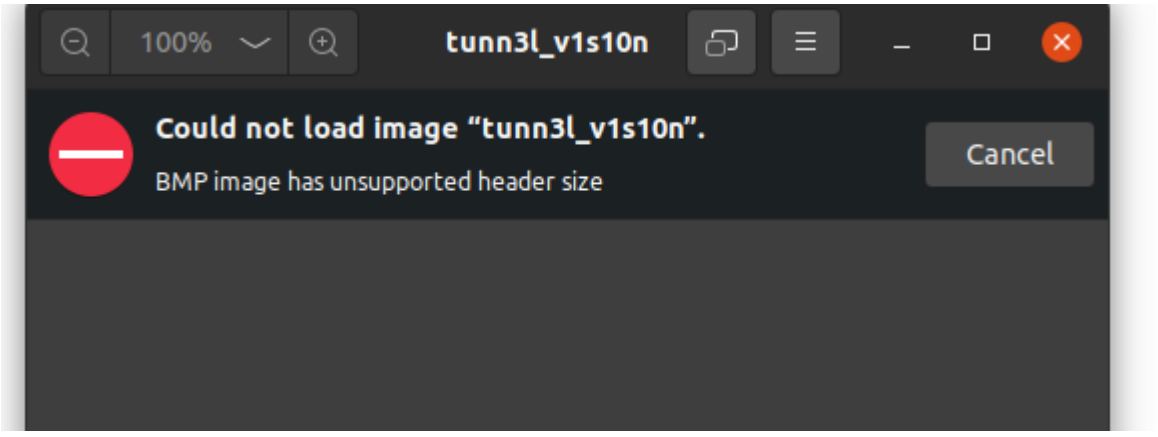
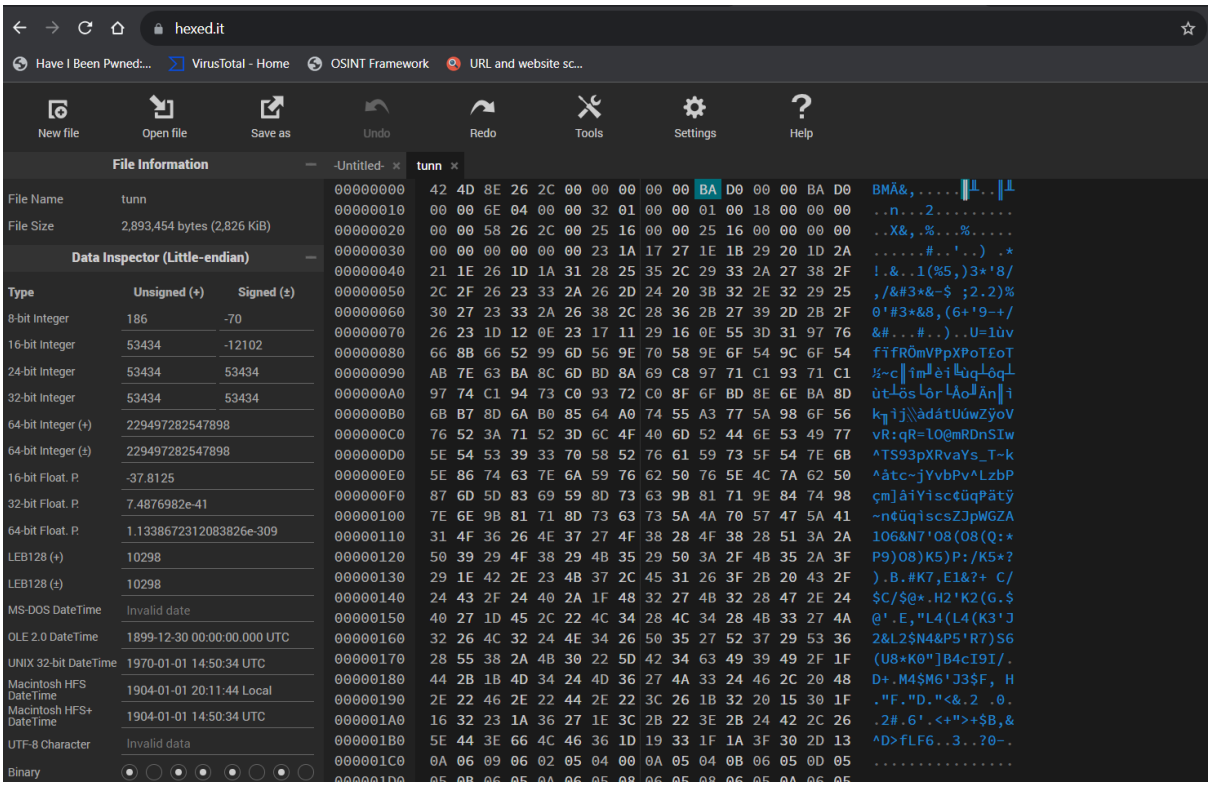
tun3l v1s10n:

A bmp file is downloaded from the picoCTF page.





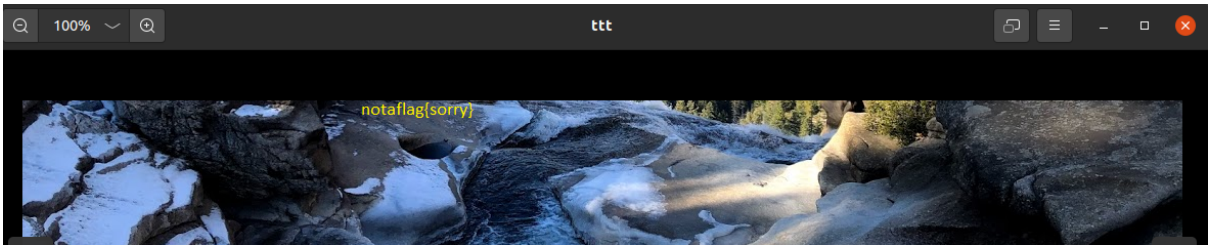
So we have to adjust the size of a bmp image. This can be done by altering the hexdump of said file.



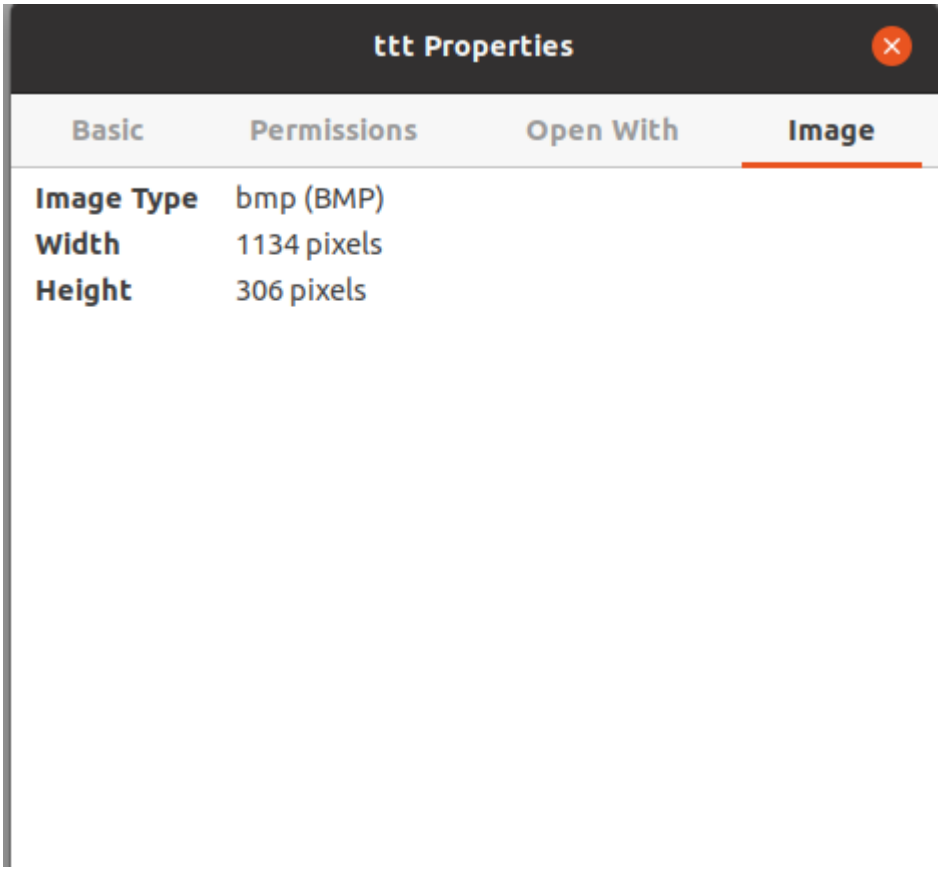
Upon comparing this to a standard bmp file, I found that

Are supposed to have static values (its also written **BA D0** to indicate that this is where changes have to be made). So I changed them.

This is what I got:



So there is something more hidden in the image.

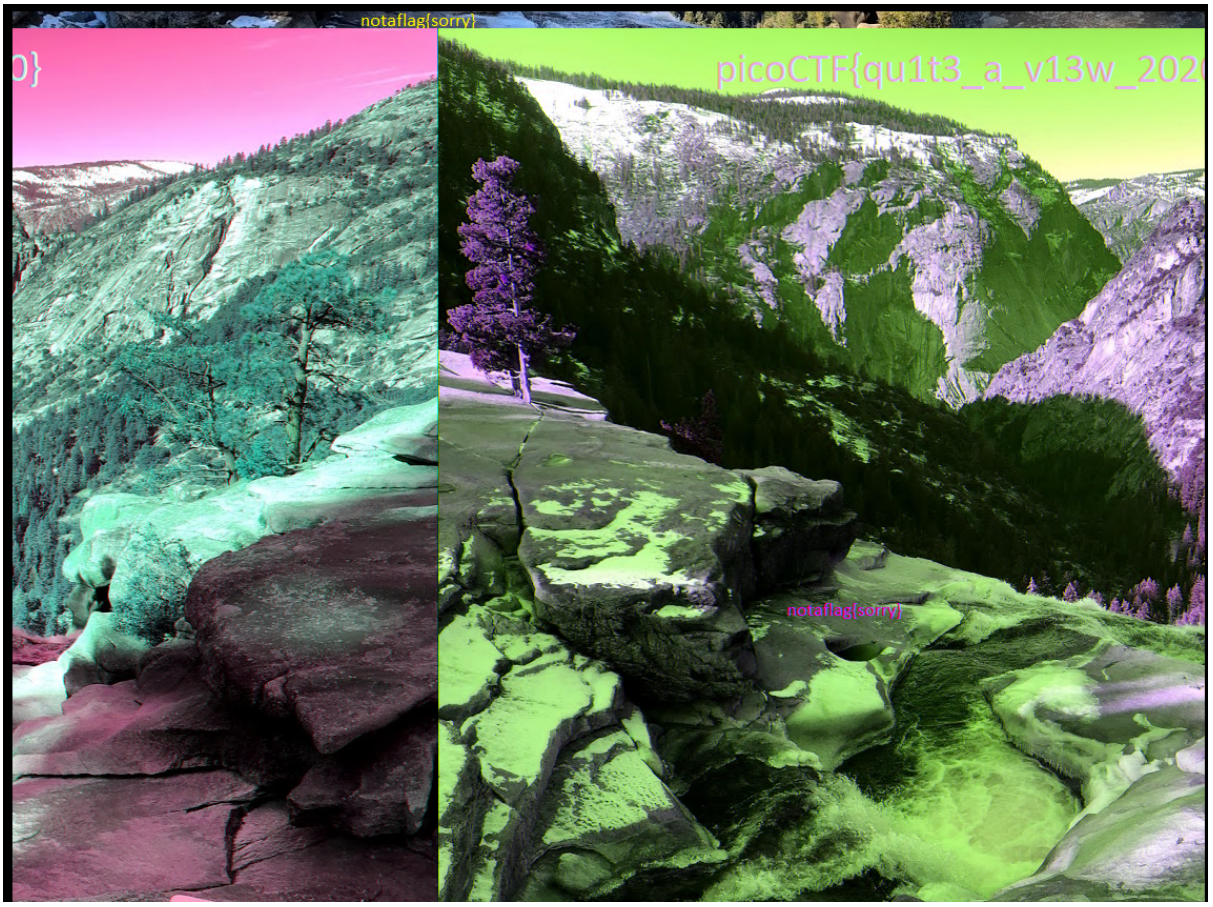


Let us try extending the height, as it is too less in comparison with the width and might contain some extra clues.

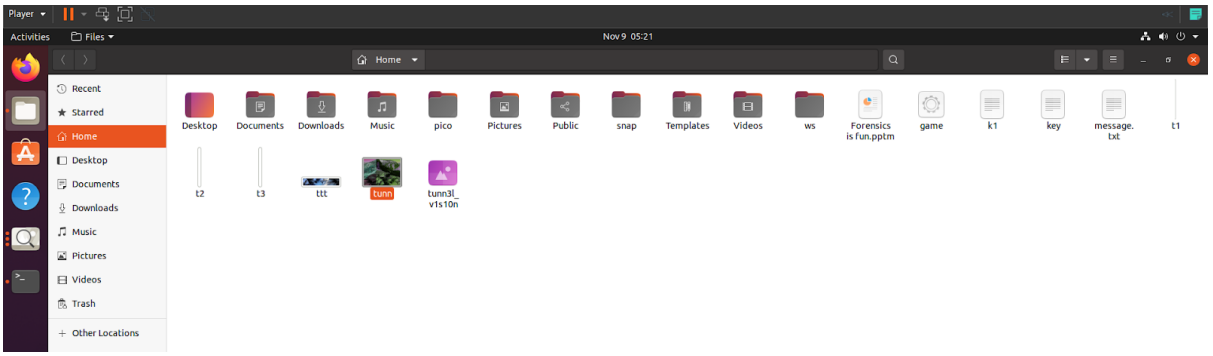
The maximum permissible height in 32K pixels for a bmp image . I tried 2000 pixels at first. This

made the image too big to view anything. Same with 1000 pixels. But when I tried it out with 850 pixels.

I did this by converting the pixel lengths into their equivalent hexadecimal values in a python script (same one I used in an earlier CTF) and replacing them in the hexdump by comparing with hexdump of a standard file.



We have our flag.



(t1,t2,t3 are the file with height 2000,1000,900 pixels respectively)

[Sources: StackOverflow, hexEd.it, SuperUser, Online BitMap Resizer]