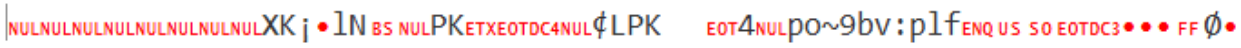
To solve this challenge, I first uploaded this file on cyberchef since, as the title implies, I will likely need to xor the file with something.



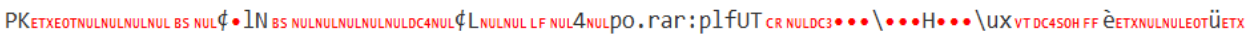
Looking at the header, I conclude 3 three things:

1) The file was most likely xored with the first 8 bytes of the header

2) The file was most likely a zip given the PK we see

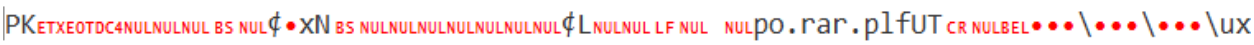
3) the file contained is currently po_9bv:plf, we need to keep an eye on it's name.

After xoring with 50 4B 03 04 00 00 00 00 the header becomes:



The first 4 bytes of zip are always the same. Bytes 5 and 6 correspond to the version. Bytes 7 and 8 correspond to any special flags regarding encryption/compression modifiers.

After a few attempts, I settled for version 2.0, A.K.A 50 4B 03 04 14 00 00 00



Finding the last 2 bytes could be done through bruteforce, especially since the 8th byte is likely 00 anyway. However, I notice that filename is po.rar.plf. However, this is very similar to xo.rar.pdf, an expected possible file name. After a little test, "p" xor "x" and "l" xor "d" are both 08, so I test out 50 4B 03 04 14 00 08 00 and… the file is restored.

The file appears to be empty, however a ctrl+a reveals hidden text that with ctrl+c and ctrl+v somewhere else reveals the contents of the pdf is the flag.

Made with love by: AndreiCat