

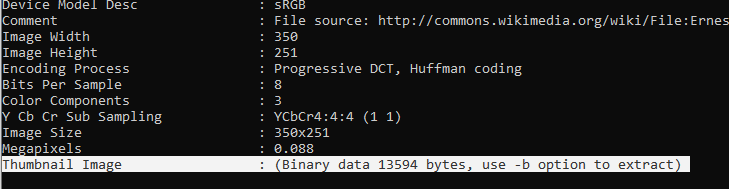
**Challenge 1:**



Notes:

Key components of the image are the image itself and the writing on the image which looks potentially like an encryption key.

1. To see if anything else is happening inside the image exiftool was used to analyse the image.
2. The image contained a link to its source (a Wikipedia page <http://commons.wikimedia.org/wiki/File:Ernest_Rutherford_1905.jpg>) so I downloaded that original image to compare in exiftool. Not much stood out.



1. At this point I went to the Wikipedia article and read up on Ernest Rutherford. I found the section where the image was taken from, a section about his discovery of the Rutherford model of the atom (which had a notable focus on the discovery of an interior).
2. Next I extracted the binary thumbnail (*exiftool.exe c.jpg –b*). It looked like an ordinary file. Extracting it as an image proved similar (*exiftool.exe -a -b -W NEW/%f\_%t%-c.%s -preview:all c.jpg*).
3. I then inspected the binary of the image. And opened the image in gimp, where it had been edited. I also opened the image in a hex editor to see if there was anything I was missing.
4. It was time to take a look at the text written on the image.

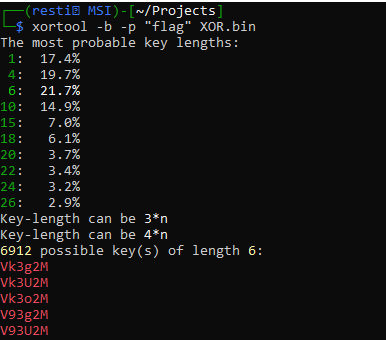
V+QQEMfkRgUXVy8d8aI93UfMI9auuIGkco2Zm7Gs2vc

+pFS1hgR7+ppKoHgyn3XeLGpUggbuAMU=

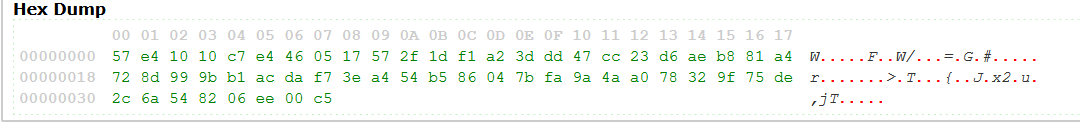
1. My first guess was that this could be a key. But because “a message on it” is listed there is a reasonable probability it is an encoded message. The first line was 44 characters long, the second line was 34. Another plausible option is that either line could be a hash code. They aren’t quite formatted correctly though.

Since + and = are the only symbols we likely are either using base64 or they hold some significance as separators and delimiters.

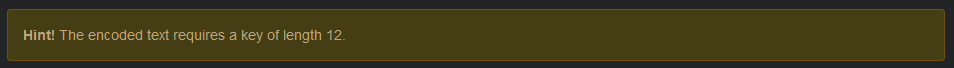
1. At this point I decided to take another approach and ran xortool, a powerful cypher decryption tool on the two lines given.



1. I further consider the option of the words being base64 (any smaller base system ruled out because of the number of characters). I run through some conversions. Base64 to plaintext/hex appears mostly garbage. An Xor bruteforce attack doesn’t return any results.



1. A clue given for the challenge lists the encoded text requires a key of length 12. This makes me consider XOR further.



*Incomplete…*