Crypto 0x003

Jonathan Ho edited this page on Mar 24, 2020 · 3 revisions

Challenge

crypto 0x0003

150

I got loyalty, got royalty inside my...

author:- Mr.7i74N

Cryptochall3...

Flag

Submit

For this challenge, we are given a txt file.

Link: https://github.com/joeyjon123/riftCTF/blob/master/Cryptochall3.txt

The file contains the following string:
"GATCCGGCGCGCACTCTAACACCCGCACTGTCTACCTCTAACTACGTCTTCCTATCCAGCGCGCCCGCTTCCGCGCGATCAATACTGCTTCCTAACAATAGATCTTGCGTACACTACTTC".

From our biology lessons, we recognize that this is a string of DNA Codons. The hint also contained lyrics to DNA, a song/rap by Kendrick Lamar.

Looking up DNA encryption/decryption techniques, we find that A, C, G, and T are all mapped to binary values. When combined, they form a binary string which should decode to the flag.

We first tried using the standard mappings: A(0) – 00 | T(1) – 01 | C(2) – 10 | G(3) – 11

Resulting binary string:
1100011010111110111011100010011001000010001010101110001001110110010010100110010000100100101101100101101001000110100011101110111010101110010110101110111011000110000001001001111001011010010000100000010011000110010111101101001000100010010010110

When we translate this to ASCII, we do not get a valid string. Therefore, we decided to start with the flag.

We know the flag begins with riftCTF{ and the first letter is r. Converted to binary, r is 01110010. The first 4 characters of the string we were given are GATC.

Therefore, we tried to solve using these mappings: T(0) – 00 | G(1) – 01 | C(2) – 10 | A(3) – 11

Resulting binary string:
0111001010010110011001101110001000111110111010100110111000010010001110100010001111100011100100100000101000110010101101100110011010100110000010100110011001110010111100111000011000001010001111101111001101110010000011110000111000010000010000001001100010010111110110100100010010010010110

When we translate this to ASCII, we do not get a valid string again. Thus, we decided to reverse every block of 4 characters.

Resulting string:
CTAGCGGCCGCGCTCACAATCCCACACGCTGTCCATATCTATCACTGCCCTTCTATCGACCGCGCGCCCCTTCGCGCTAGATAACGTCCCTTCAATATAACTAGCGTTCATGATCACTTC

Repeating the steps as before, we know that CTAG maps to r which maps to 01110010. We figure out that the mappings for each character are: A(0) – 00 | C(1) – 01 | G(2) – 10 | T(3) – 11

Resulting binary string:
0111001001101001011001100111010001000011010101000100011001111011010100110011011100110100011110010101111101110011011000010110011001100101010111110110011001110010001100000110110101011111010000110011000001110010011011110100111000110100011111101

When we translate this to ASCII, we get riftCTF{S74y_safe_fr0m_C0roN4}.

Flag: riftCTF{S74y_safe_fr0m_C0roN4}

Pages 23

Find a page...

Home
Crypto 0x001
Crypto 0x002
Crypto 0x003
Crypto 0x004
Crypto 0x005
Crypto 0x007
Misc 0x001
Misc 0x002
Misc 0x003
Misc 0x004
Misc 0x005
Misc 0x006
Osint 0x001
Osint 0x003

Show 8 more pages...

Clone this wiki locally

https://github.com/csn3rd/riftCT