The program read data from `Flags.txt` then encode the flag to become symbol string. We remove all unnecessary code like: print the mask, remove flag file.

devil\_function(): do nothing

zarathos(): use rand() to find the index to reverse the string. Its reverse 3 times.

Lucifer(): use rand() as index of array to add or sub the value of “reverse strings”.

* Adfedd(): its change hex valule to dec. Example: K -> 75(dec) -> 75(ascii) -> 37 35(hex)

matter\_manipulation[abi:cxx11](): change strings that’s Lucifer() create as mapping array. Each mapping value locate in the program, we could file it when using “xref to std::map”. There 10 value for mapped 0-9.

Since the challenge use rand() a lot, we cant predict how the flag reverse the decrypt correctly. We have to bruceforce => index for reverse string isnt the problem anymore.

symbols = ['\*', ')', '$', '@', '^', '#', '(', '!', '%', '&']

with open('Your\_hacked\_data0.txt', 'r') as file:

    data\_array = [line.strip() for line in file]

def replace\_symbols(data, symbols):

    for i in range(len(data)):

        for symbol in symbols:

            if symbol in data[i]:

                data[i] = data[i].replace(symbol, str(symbols.index(symbol)))

    return data

def numbers\_to\_ascii(number\_string):

    for j in range(30):

        pairs = [(int(number\_string[i:i+2]) + j) for i in range(0, len(number\_string), 2)]

        ascii\_string = ''.join(chr(pair) for pair in pairs)

        if "VISHWACTF" in ascii\_string:

            print(ascii\_string)

data\_array = replace\_symbols(data\_array, symbols)

for data in data\_array:

    numbers\_to\_ascii(data)

We found the flag: VISHWACTF[<\_4R3\_R4N50MW4R35\_B3AUTIFUL=?\_>[:)]] but cant submit => open ticket and we got reply: “flags don't contain special chars in general except for \_ “

* Flag: VISHWACTF[4R3\_R4N50MW4R35\_B3AUTIFUL]