THIS BLOWS

Description:

Looks like my code has some bugs. I was able to get it encoded but now I can't get back to the flag.

Encoded.txt

```
NDMxODcyZWNkYjQzNDM2YWVhNzNmNTgzZGE4NWExNTk4ZTJjYWQ5ZDk1NDUwOWQ3M2RlNjE4ZWM2ZjNl
YjlmNjUxNjgyZWFlYjc4N2UyODhkMjE4ZGI4NTlhNGFkYWE1
```

i.txt:

```
MzBCQzRGQUE2OUNGRjEw
```

k.txt:

```
QTBGRkRFMTI=
```

Decode.py

```
#!/usr/bin/env python
import base64
from Crypto.Cipher import Blowfish
from Crypto.Random import get_random_bytes

a23 = 0
b78 = 0
c45 = 0

# To Do: fix the broken decode function
def decode(string):
    e = string
    k = a2
    i = b78

    c = Blowfish.new(k, Blowfish.MODE_CBC, i)
    #h = cipher.decrypt(ciphertext)

    #print(h)

# Parse files, possible issue
def parse():
    with open('k.txt', 'r') as f:
        k = f.read()

    a23 = b64(k) # a23 is a global variable which is decoded from b64

    with open('i.txt', 'r') as g:
        i = g.read()

    b78 = i

    with open('encoded.txt', 'r') as h:
        z = h.read()

    c45 = b64(z)

# Decodes b64 to raw
def b64(string):
    d1 = base64.b64encode(string)
    return d1

def main():
    #parse
    parse()
```

```
    #decodes parsed data
    decode(c45)
    fishfossil()

#Runs the main function
if __name__ == "__main__":
    main()
```

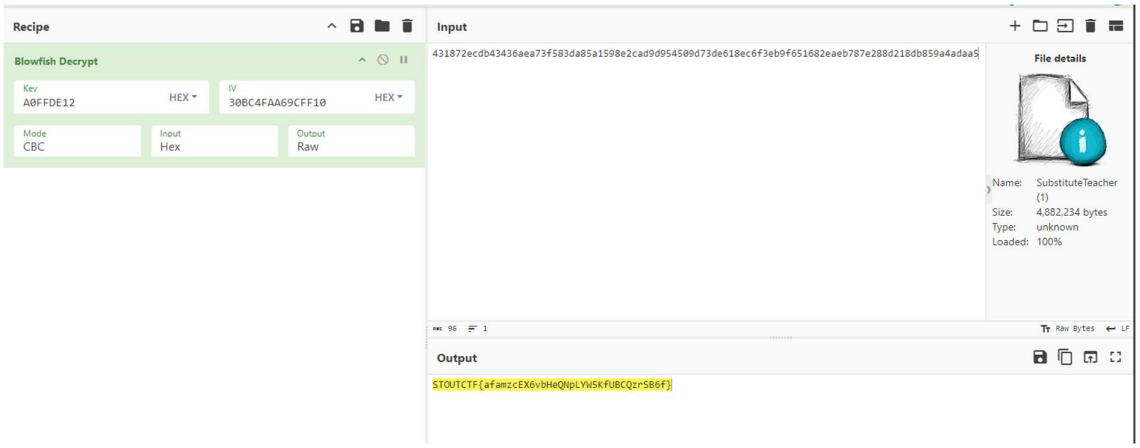
Reading through this line below see the encoded uses Blowfish hence we can decode it by scripting or directly use cyberchef for convenience. **The *i.txt* and *k.txt* must be convert from base64 first the put the Hex value on the details.**

```
c = Blowfish.new(k, Blowfish.MODE_CBC, i)
```

Script

```
from Crypto.Cipher import Blowfish
from Crypto.Util.Padding import unpad
import binascii

key = b'\xA0\xFF\xDE\x12' #FROM BASE64 TO HEX TO BYTES
iv = b'\x30\xBC\x4F\xAA\x69\xCF\xF1\x00' #FROM BASE64 TO HEX TO BYTES
encoded_hex =
"431872ecdb43436aea73f583da85a1598e2cad9d954509d73de618ec6f3eb9f651682eaeb787e28
8d218db859a4adaa5" #FROM BASE64

encoded = binascii.unhexlify(encoded_hex)

cipher = Blowfish.new(key, Blowfish.MODE_CBC, iv)

cdec = cipher.decrypt(encoded)

try:
    flag = unpad(cdec, Blowfish.block_size).decode('utf-8')
except ValueError:
    flag = cdec

print(flag)
```

ALTERNATIVE WAY:



Flag	STOUTCTF{afamzcEX6vbHeQNPLYWSKFUBCQzr5B6f}