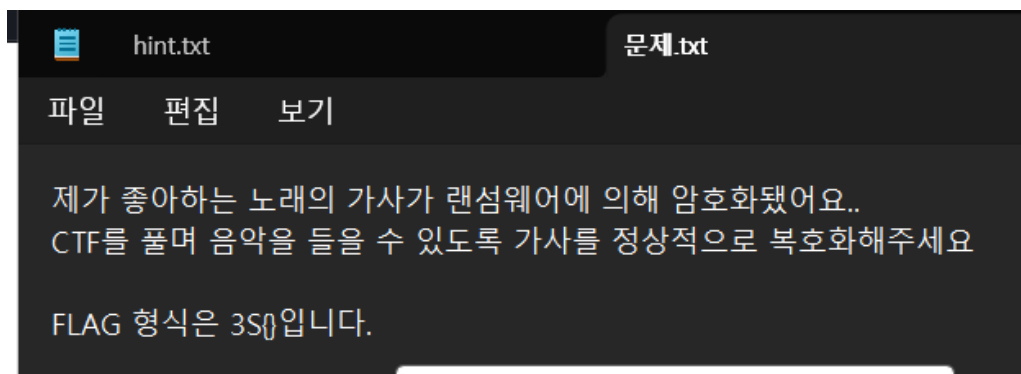


what's This Song

32기 김효주

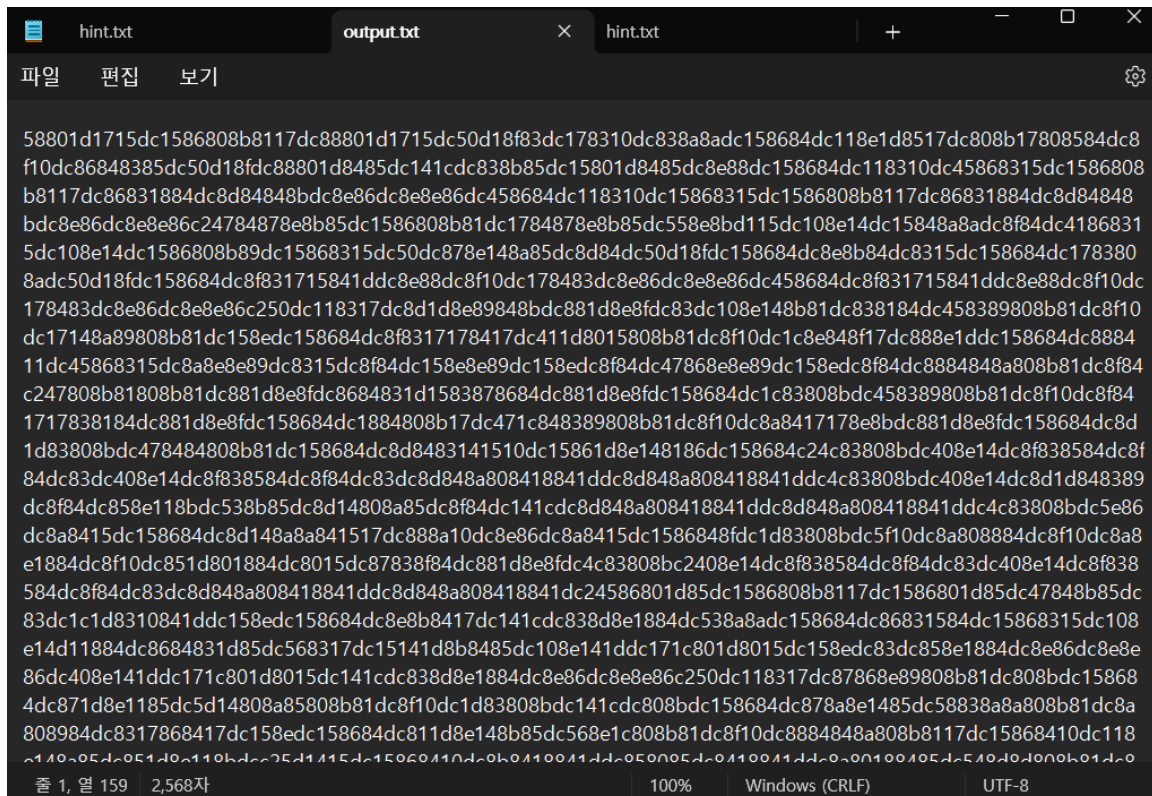
크립토 문제도 처음 풀어본다

문제는 다음과 같다 복호화 문제군



Output 파일엔 이런 게 있다

이걸 복호화하면 플래그를 찾을 수 있는 것 같다



위키백과

우리 모두의 백사사전

≡

X

≡ 빈도분석 (암호)

문서 토론

위키백과, 우리 모두의 백사사전.

검색 ∞ 계정 만들기 로그인 ...

국가 예디터이온 8월 14일부터 9월 1일까지 진행됩니다.

위키백과

우리 모두의 백사사전

≡

X

≡ ASCII

문서 토론

위키백과, 우리 모두의 백사사전.

검색 ∞ 계정 만들기 로그인 ...

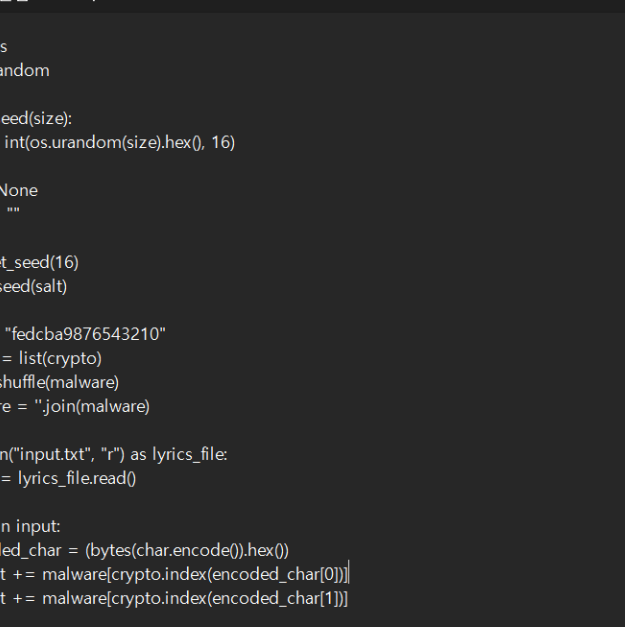
국가 예디터이온 8월 14일부터 9월 1일까지 진행됩니다.

알파학에서의 **빈도분석**(頻度分析, frequency analysis 또는 counting letters)이란 **문론**과 **암호론**에 사용되는 문자 또는 문자들의 출현빈도를 단서로 이용하는 암호해독법을 말한다. 평문 언어의 통계적 특징을 전제로 하여, 암호문만을 사용해서 해독을 진행한다 때문에, 암호문 단독복각으로 분류된다.

개요 [편집]

평문 한 글자를 다른 글자(또는 숫자나 기호 따위)로 1대1로 변환해서 암호문을 작성하는 **단일 알파식 암호**에는, 평문과 암호문으로 대응하는 문자의 출현빈도가 일치한다는 특징이 있다. 일반적으로, 평문 문자들은 특정 문자에 치우쳐 있으며, 문장에 상관 없이 거의 일정한 양이 나오며, 평문 문자의 출현빈도와 암호문 문자의 출현빈도를 대조하는 것으로 평문의 복각가능해 제어 무자통과 공격을 비로

정리해보자면

- 
- ```
import os
import random

def get_seed(size):
 return int(os.urandom(size).hex(), 16)

input = None
output = ""

salt = get_seed(16)
random.seed(salt)

crypto = "fedcba9876543210"
malware = list(crypto)
random.shuffle(malware)
malmware = ".join(malware)

with open("input.txt", "r") as lyrics_file:
 input = lyrics_file.read()

for char in input:
 encoded_char = (bytes(char.encode()).hex())
 output += malware[crypto.index(encoded_char[0])]
 output += malware[crypto.index(encoded_char[1])]

with open("output.txt", "w") as result_file:
 result_file.write(str(output))
```

```
try:
 first_char = malware_to_crypto[hex_pair[0]]
 second_char = malware_to_crypto[hex_pair[1]]
except KeyError:
 print(f"Error: Invalid character in encrypted text: {hex_pair}")
 return None

hex_value = first_char + second_char
decrypted_text += chr(int(hex_value, 16))

i += 2

m_decrypted_text

d_text = ""
58801d715dc15680f0eb811cd88801d715dc04af863dc178310dc838a
= "031f9b2a4e67d85c"
"fedcba9876543210"

d_text = decrypt_lyrics(encrypted_text, malware, crypto)

crypto_text)
```