

# Secret Document

문제

462명 해결


✕


Secret Document  
150


Author: Chu

 Flag.zip

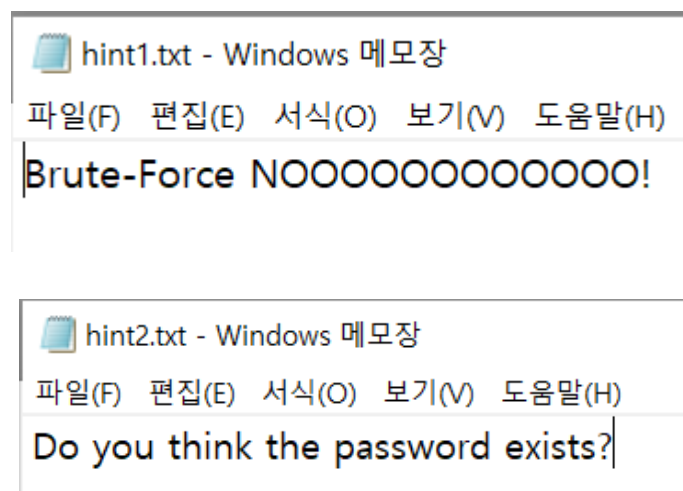
제출

 flag.txt\*

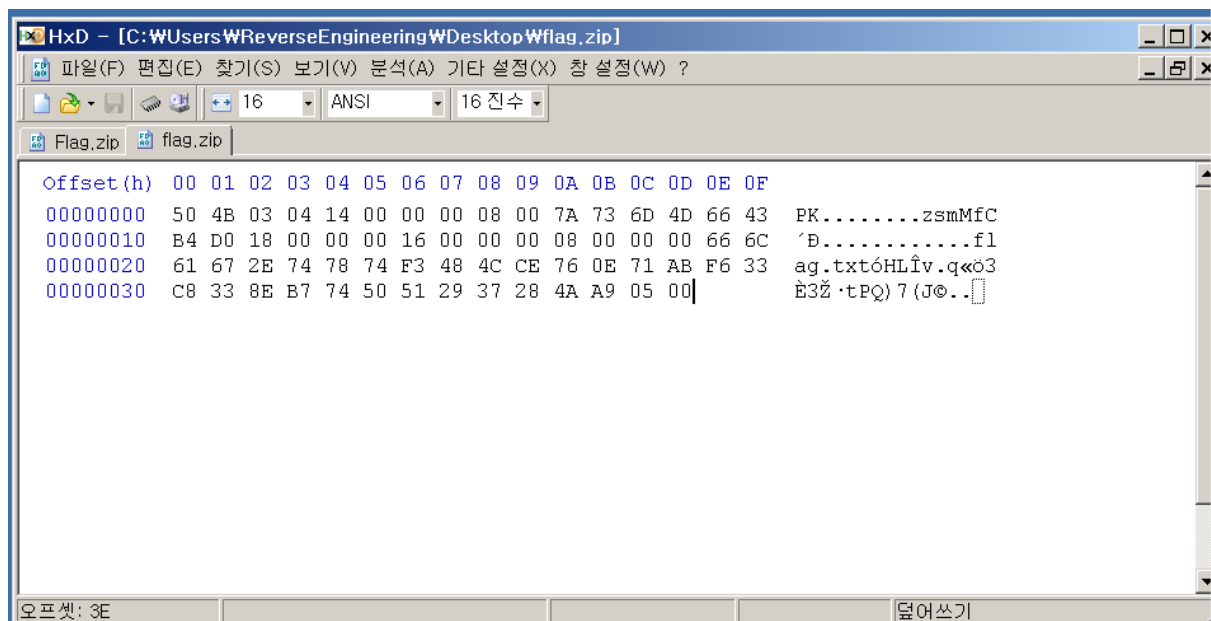
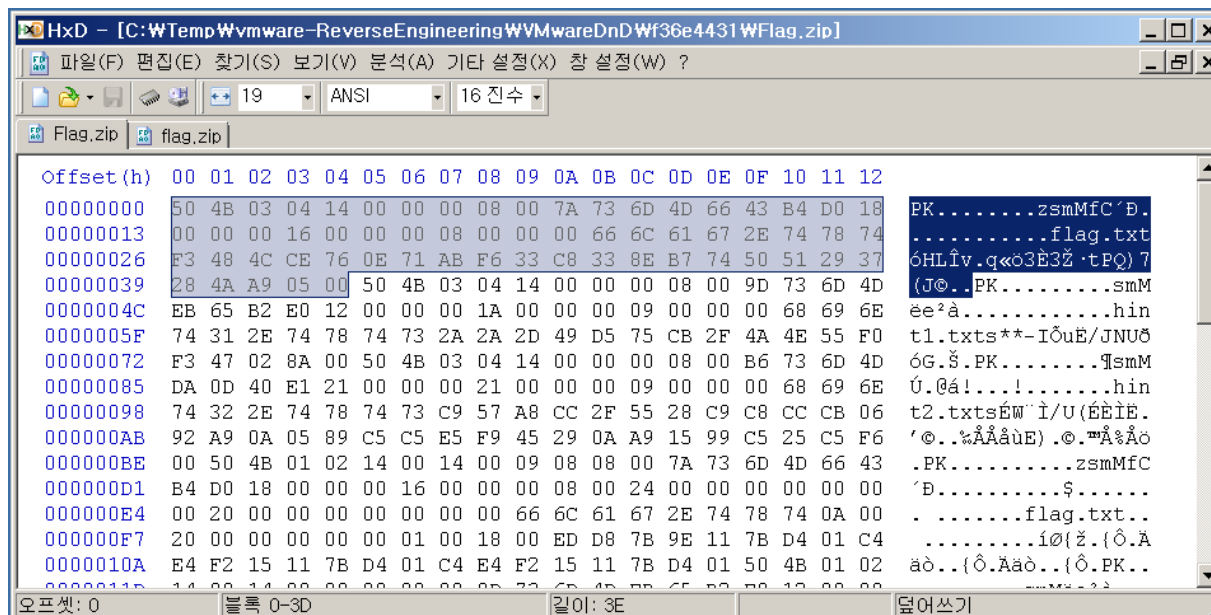
 hint1.txt

 hint2.txt

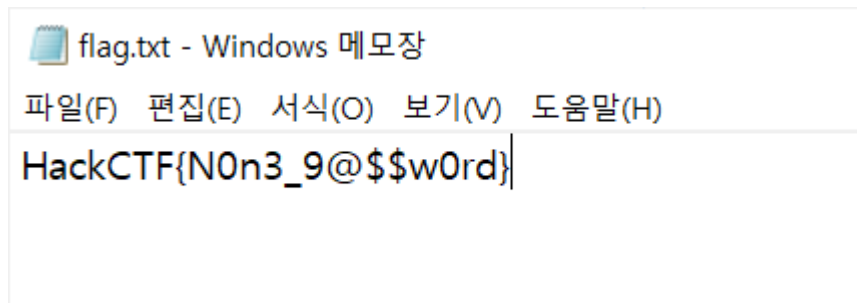
압축 파일을 열어보면 이 3가지 파일이 들어있고, flag.txt에는 암호가 걸려있다.



password를 찾는 문제는 아닌 것 같으니 압축파일을 일단 HxD로 열어보았다.



zip파일 헤더 시그니처가 여러번 반복해서 나오길래 flag.txt가 들어있는 부분에서 다음 헤더 시그니처가 나오기 전까지의 부분을 발췌해 zip파일 새로 생성해보았다.

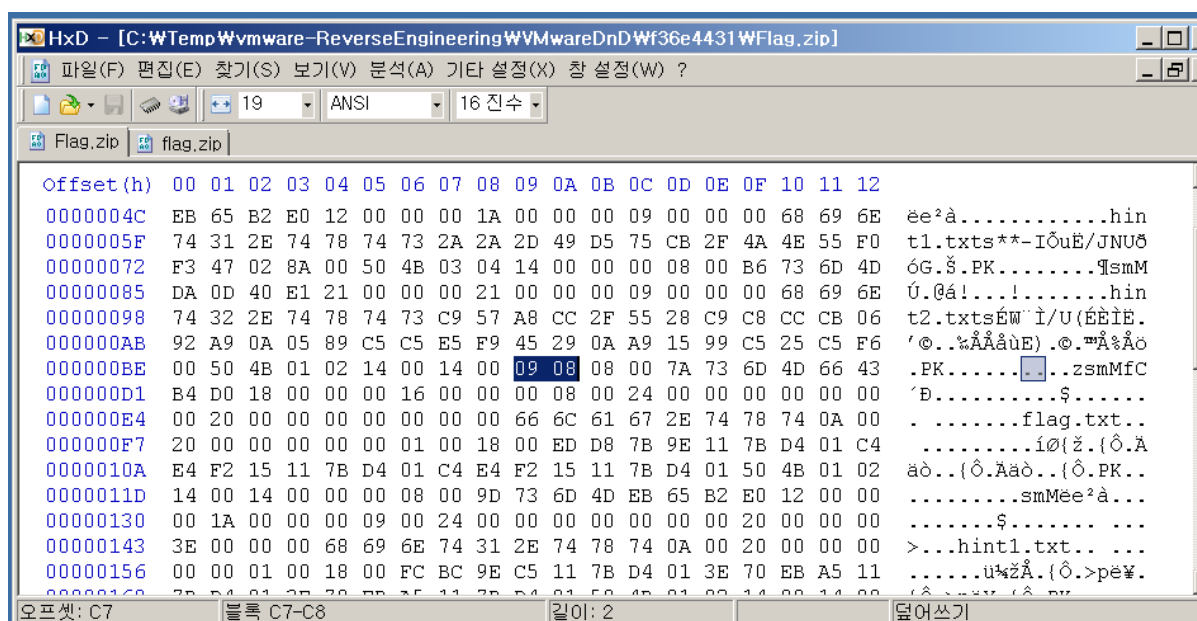


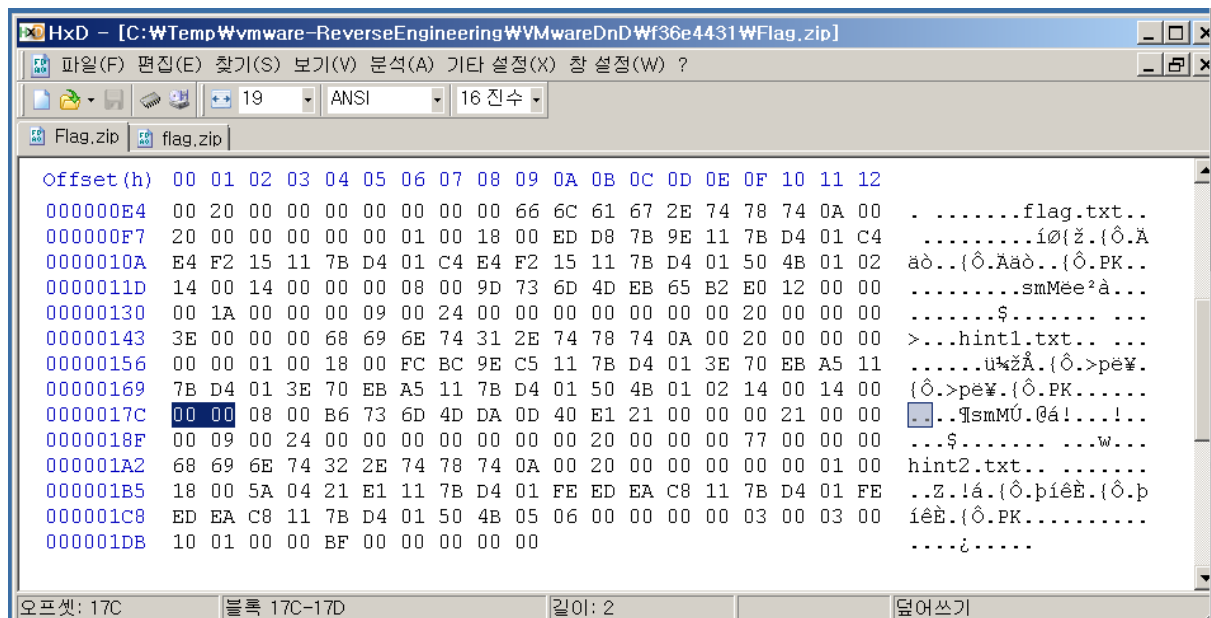
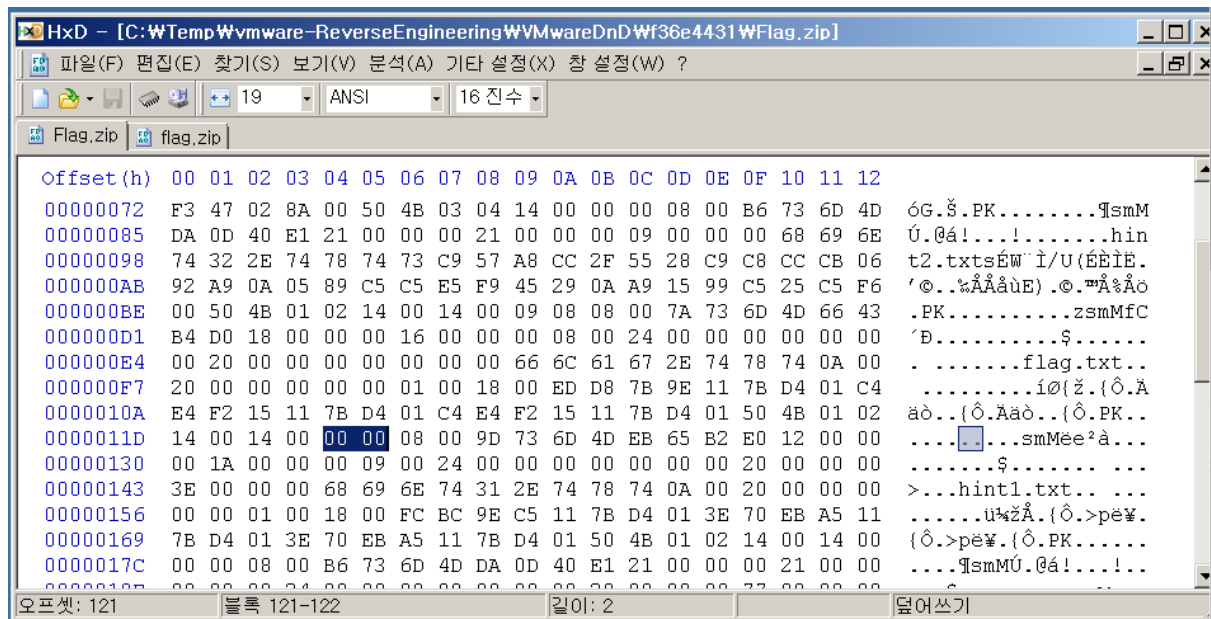
이렇게 flag를 얻긴했는데 찾아보니 이런 방식으로 푸는 게 아닌 것 같다.

zip 파일의 구조를 살펴보면 아래 그림과 같은데

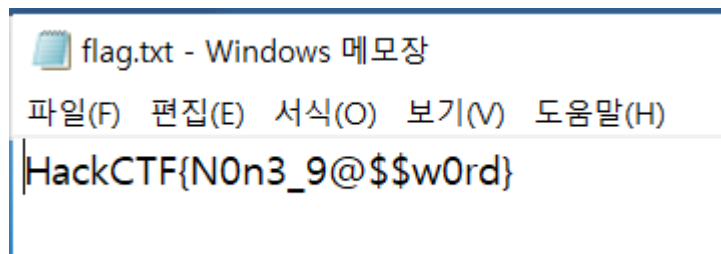
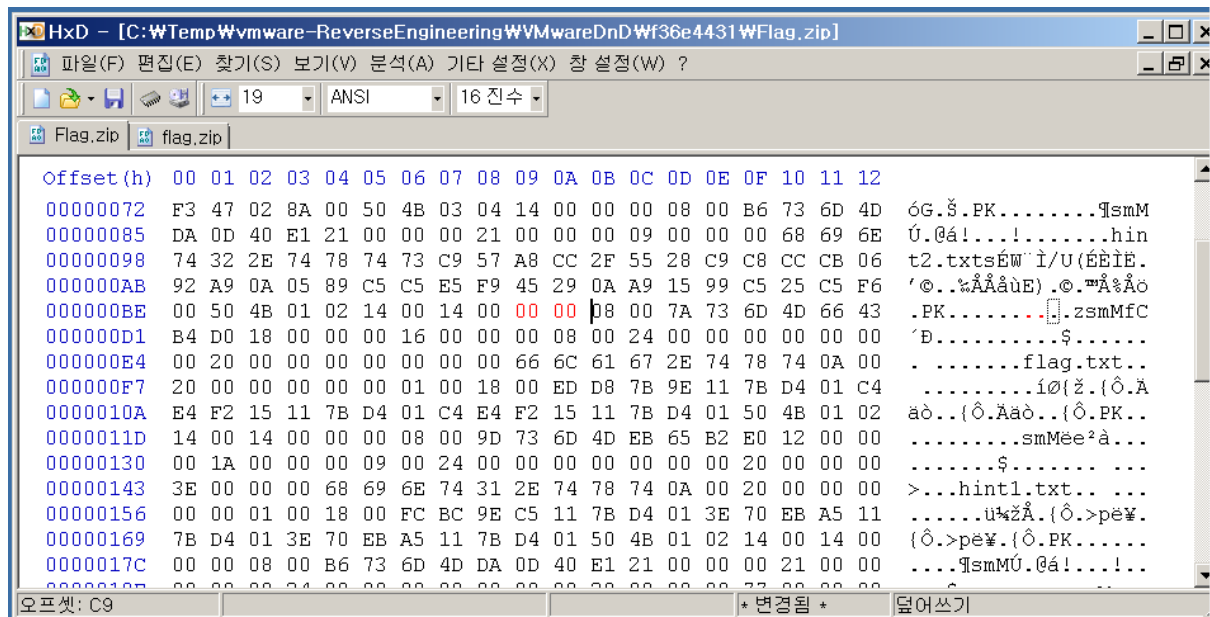
	0x0	0x1	0x2	0x3	0x4	0x5	0x6	0x7	0x8	0x9	0xa	0xb	0xc	0xd	0xe	0xf
0x0000	Signature				Version		Vers. needed		Flags		Compression		Mod:time		Mod:date	
0x0010	Crc-32				Compressed size				Uncompressed size				File name len		Extra field len	
0x0020	File comm. len		Disk # start		Internal attr.		External attr.				Offset of local header					
0x0030	File name:(variable)															
0x0040	Extra field (variable)															
0x0050	File comment (variable)															

flag는 0x8, 0x9부분에 위치해있다고 하니 이부분을 살펴보겠다.





flag.txt만 09 08으로 되어있고 hint1.txt, hint2.txt에는 00 00 으로 되어있으니 flag.txt부분  
도 00 00으로 바꿔주자!



+ 앞에서는 hin1.txt, hint2.txt부분을 보고 끼워맞춘거고, 이부분에 대해 좀 더 알아보자.  
bit 플래그를 살펴보면

## Flags

General purpose bit flag:

- Bit 00: encrypted file
- Bit 01: compression option
- Bit 02: compression option
- Bit 03: data descriptor
- Bit 04: enhanced deflation
- Bit 05: compressed patched data
- Bit 06: strong encryption
- Bit 07-10: unused
- Bit 11: language encoding
- Bit 12: reserved
- Bit 13: mask header values
- Bit 14-15: reserved

암호화와 관련된 비트가 00, 06이다.

flag.txt의 flag가 09 08 이었는데 이를 리틀엔디안 방식으로 보면 08 09가 되고 bit별로 값을 보면

값	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	1
bit	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00

bit 00이 1로 채워져있어 암호화가 된 것으로 보인다. 따라서 해당 부분을 0으로 주어 08 08으로 바꾼 후 새로 zip파일을 생성하면 flag.txt의 암호가 사라져 열람해볼 수 있다.