

Starpletech Unfile

Playing with MPQ (1) - Simple MPQ file analysis



why do you ask
2013. 10. 19. 0:16

[add neighbor](#)

Protect is linked to unprotect.

If you know how to protect, you can also unprotect.

This tutorial is not intended to actually protect your maps,

It is possible to protect in this way, and so on.

The target of the course is aimed at those who can do basic programming.

The standard of 'basic' is whether you can make Tetris by yourself.

Removed boogeyum - every time it comes in, it goes crazy

Let's take a look at how MPQ files are composed.

We will proceed through a program called MPQ Helper.

MPQ Helper

- The main data of the MPQ file is encrypted. You can use mpqhelper to turn this encryption on and off again.
- MPQ files contain compression function. If you write mpqhelper, you can compress and decompress files to fit MPQ.

There are only these two main features. With only these two things, I will interpret one MPQ file.

1. Let's look at the MPQ file structure.

- Source: http://web.archive.org/web/20120222093346/http://wiki.devklog.net/index.php?title=The_MoPaQ_Archive_Format
- Source: <http://www.zezula.net/en/mpq/mpqformat.html>

whyask37's blog

When I open the file using our friend Hex Workshop, it looks like this:

A) It is very convenient that 214 comes out when you drag D600 0000. HxD may have this function, but it is cumbersome to find it. if not



whyask37's blog

MPQ 헤더 정리		
매직 넘버	"MPQ" 1Ah	4D50511A
헤더 길이	32	20000000
파일 길이	246	F6000000
MPQ 버전	스타크래프트	0000
sectorSizeShift	12	0C00
HET 오프셋	214	D6000000
BET 오프셋	230	E6000000
HET 엔트리 수	1	01000000
BET 엔트리 수	1	01000000
총	32byte	

Hmm... But I have to explain what HET or BET or sectorSizeShift is.

1) HET (Hash Entry Table)

- Hash table. There is a hash table with linear hashing in order to be able to quickly find the desired file in the MPQ file .

1. MPQ 파일 안에는 파일명이 없습니다. 파일명의 해시값만 있습니다. (이러한 이유때문에 MPQ 파일만으로는 안에 들어있는 파일 이름들을 알 수 없어서 파일명만 따로 모아 (listfile)이라고 만들어놓기도 합니다.)

2. MPQ 파일 안에는 파일명에 따른 해시 테이블이 구성되어 있습니다. 각 파일명은 각각 하나의 블록을 가르킵니다.

이차 해싱 테이블에서 각 bucket이 여기에서 entry에 해당합니다. hash entry들의 table (array)라는 뜻입니다.

자세한 설명은 HET에서 하도록 하지요.

혹시 해시 테이블을 모르신다면 자세한건 위키피디아를 참조하세요.

http://en.wikipedia.org/wiki/Hash_table

2) BET (Block Entry Table)

- 블록 테이블입니다. MPQ에서는 다음과 같이 파일을 찾아요.

파일명 -> HET -> BET -> 블록(파일 데이터)

해시 테이블의 각 엔트리들은 BET의 한 엔트리를 가르키고, BET는 각 파일 데이터를 가르키면서 각 파일의 대략적 정보를 알려줍니다.

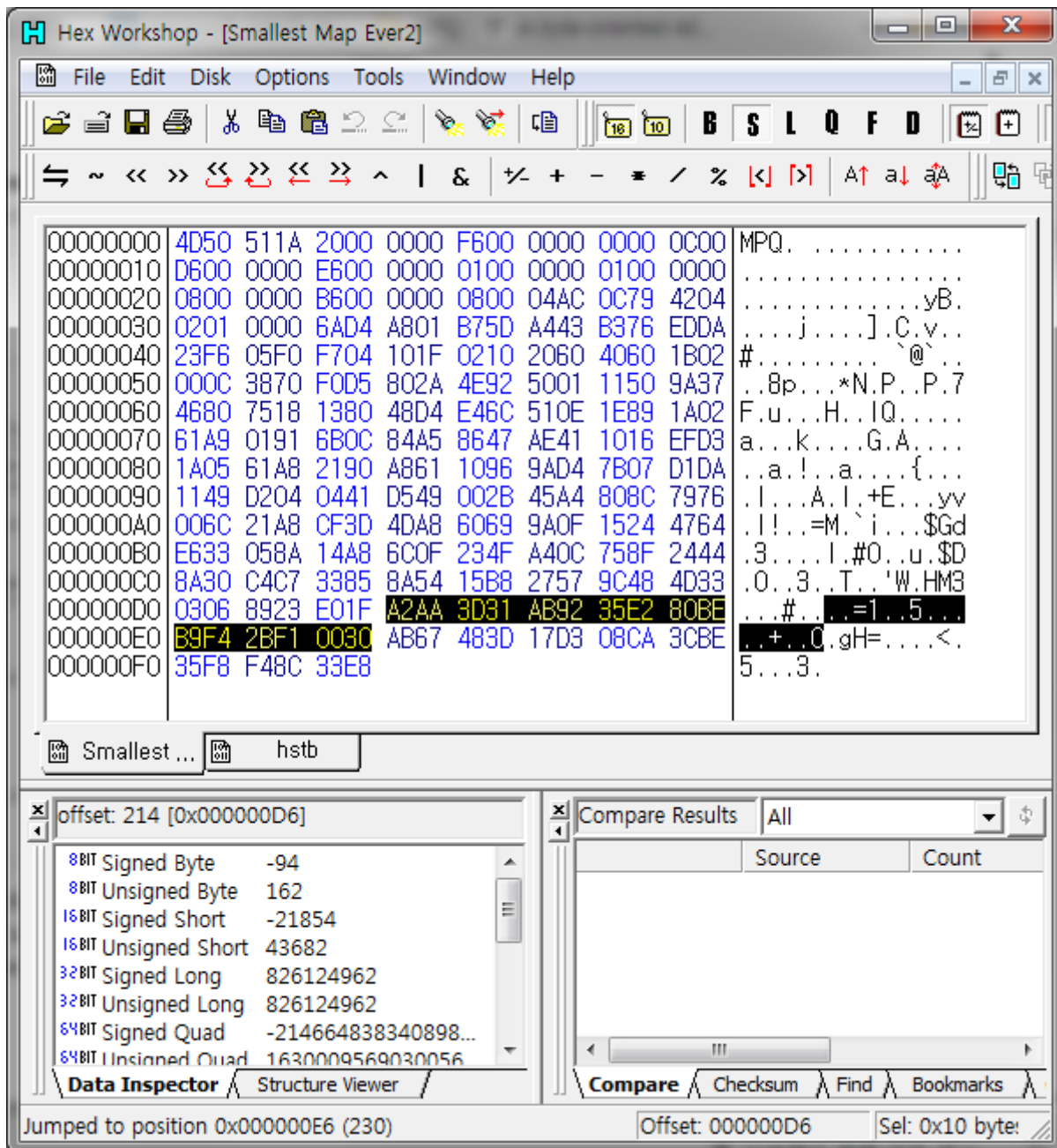
3) sectorSizeShift

- MPQ 파일 안에 저장되어있는 파일은 $1 \ll (\text{sectorSizeShift} + 9)$ 만큼의 크기 단위로 잘려져 저장되어 있습니다.

자세한 이야기는 파일 아츠의 프 레그 가너드로 하지요

whyask37's blog

1) HET 분석



여기예요. 아까 MPQ 헤더에서 HET가 파일 시작으로부터 216byte째부터 16byte (한 엔트리가 16byte 입니다. 엔트리 갯수가 1개이므로) 만큼을 차지한다 했었죠? 그래서 저렇게 블록을 선택한겁니다.

HET는 암호화되어있습니다. mpqhelper를 이용해서 암호를 풀어주는 작업부터 진행하도록 하겠습니다.

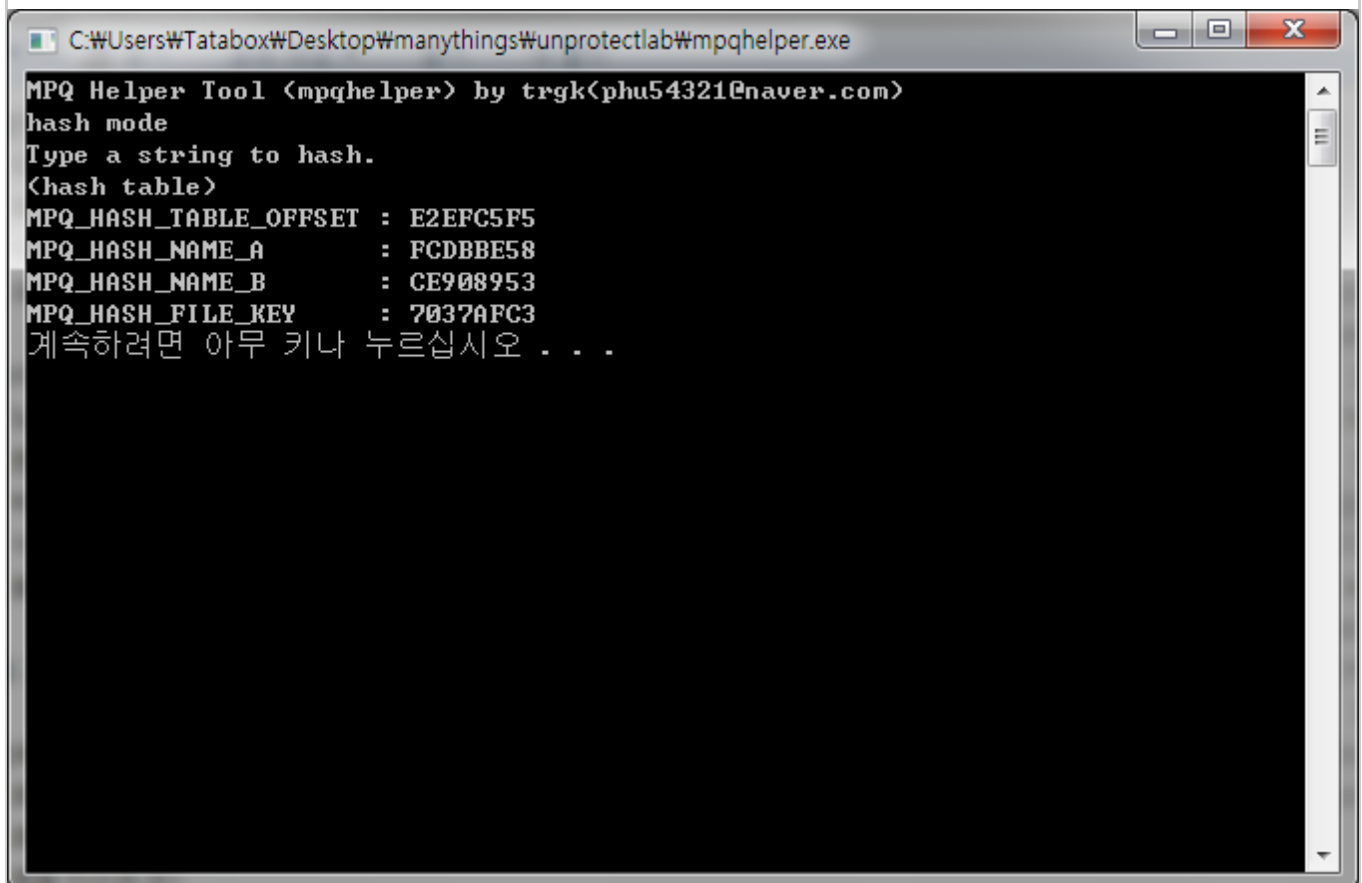
(mpqhelper 프로젝트 파일도 첨부해 올립니다. 관심 있으시면 한번 보세요. (mpqhelper.zip))
(가독성이 조금 떨어질겁니다. 뭐 짜집기해서 똑딱 만들다보니 좀 그렇네요. ㅜㅜ)

whyask37's blog

- 해쉬 계산 모드 : 그냥 실행시켰을 때
: 임의의 문자열의 해시값을 계산하게 해줍니다. 해시는 MPQ에서 사용하는 해시(이름은 없음)를 이용합니다.
- 파일 조작 모드 : cmd에서 "mpqhelper [파일명]" 처럼 실행시켰을 때
: 파일을
 - 특정 키를 사용해 MPQ에서 하는 방식으로 암호화, 복호화 (각각 enc, dec)
 - 파일을 PKWARE Compression Library의 Implode, Explode에 따라 압축을 하고 풀기도 하는 역할. (각각 imp, exp)

HET 복호화하기

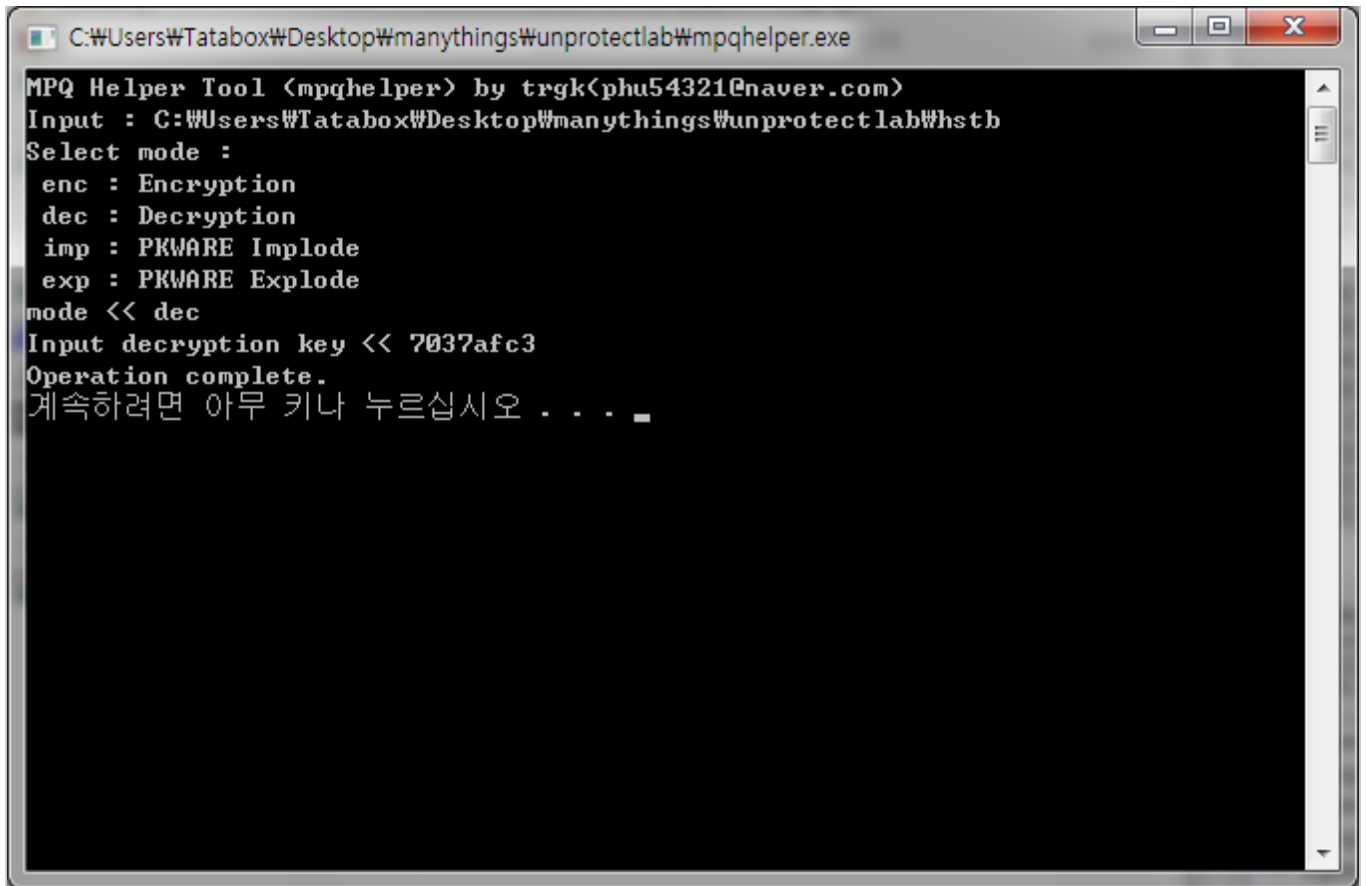
- 1. MPQ 파일에서 HET에 해당하는 부분을 따로 파일로 만듭니다. 저는 그냥 hstb로 합니다.
- 2. mpqhelper를 해시 계산 모드로 실행시켜서 "**(hash table)**"의 **해시값**을 계산시킨 후 **MPQ_HASH_FILE_KEY** 값을 읽는다.
- 3. "mpqhelper hstb" 라고 cmd에 쳐서 dec 모드(복호화)로 간 뒤에 위에서 얻은 키값을 넣어서 복호화를 시켜줍니다.
- 4. hstb.mpqdec 에 복호화된 HET가 나와있습니다.



```

C:\Users\Tatabox\Desktop\manythings\unprotectlab\mpqhelper.exe
MPQ Helper Tool <mpqhelper> by trgk<phu54321@naver.com>
hash mode
Type a string to hash.
(hash table)
MPQ_HASH_TABLE_OFFSET : E2EFC5F5
MPQ_HASH_NAME_A       : FCDBBE58
MPQ_HASH_NAME_B       : CE908953
MPQ_HASH_FILE_KEY     : 7037AFC3
계속하려면 아무 키나 누르십시오 . . .
  
```

과정 사진 1 - (hash table)의 해쉬값 얻기

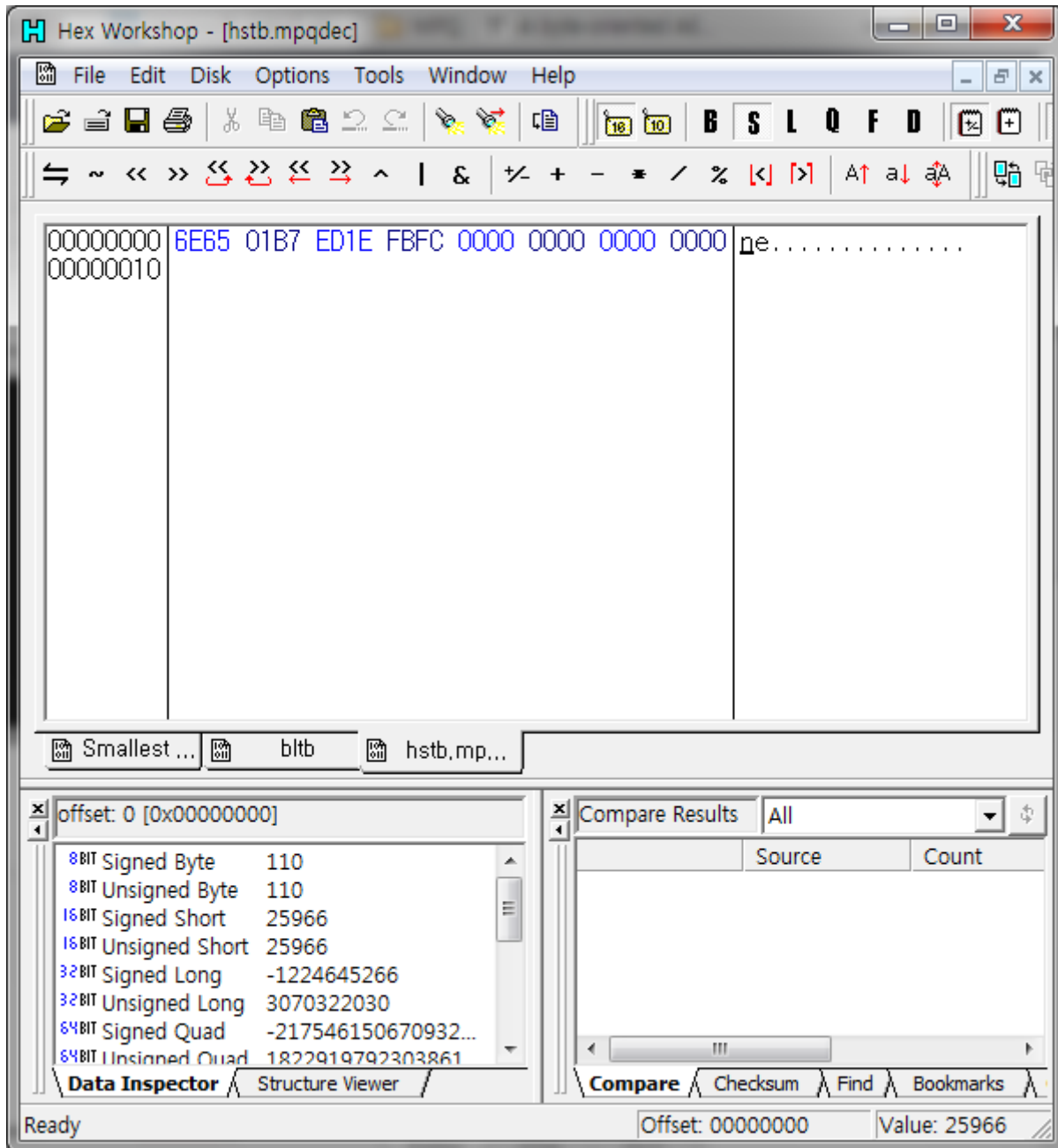


```
C:\Users\Tatabox\Desktop\manythings\unprotectlab\mpqhelper.exe
MPQ Helper Tool <mpqhelper> by trgk<phu54321@naver.com>
Input : C:\Users\Tatabox\Desktop\manythings\unprotectlab\hstb
Select mode :
  enc : Encryption
  dec : Decryption
  imp : PKWARE Implode
  exp : PKWARE Explode
mode << dec
Input decryption key << 7037afc3
Operation complete.
계속하려면 아무 키나 누르십시오 . . .
```

과정 사진 2 - hstb 복호화

주의 : 저기 나오는 키값이나 해시값들은 모두 리틀엔디안처럼 표시하도록 되어있습니다. 그게 읽기 편해요.
실제 해시값이나 키값은 바이트 순서를 원래대로 다시 바꿔줘야 합니다;

이렇게 hstb를 복호화해서 다음과 같이 됩니다.



6E65 01B7 ED1E FBFC 0000 0000 0000 0000

HET 정리			
엔트리 #0	hashA	오른쪽값	6E6501B7
	hashB	오른쪽값	ED1EFBFC
	언어	Default	0000
	플랫폼	0(Default)	00
	BET 인덱스	0	00000000
	총	16byte	
총 엔트리	1개	총 크기	16byte

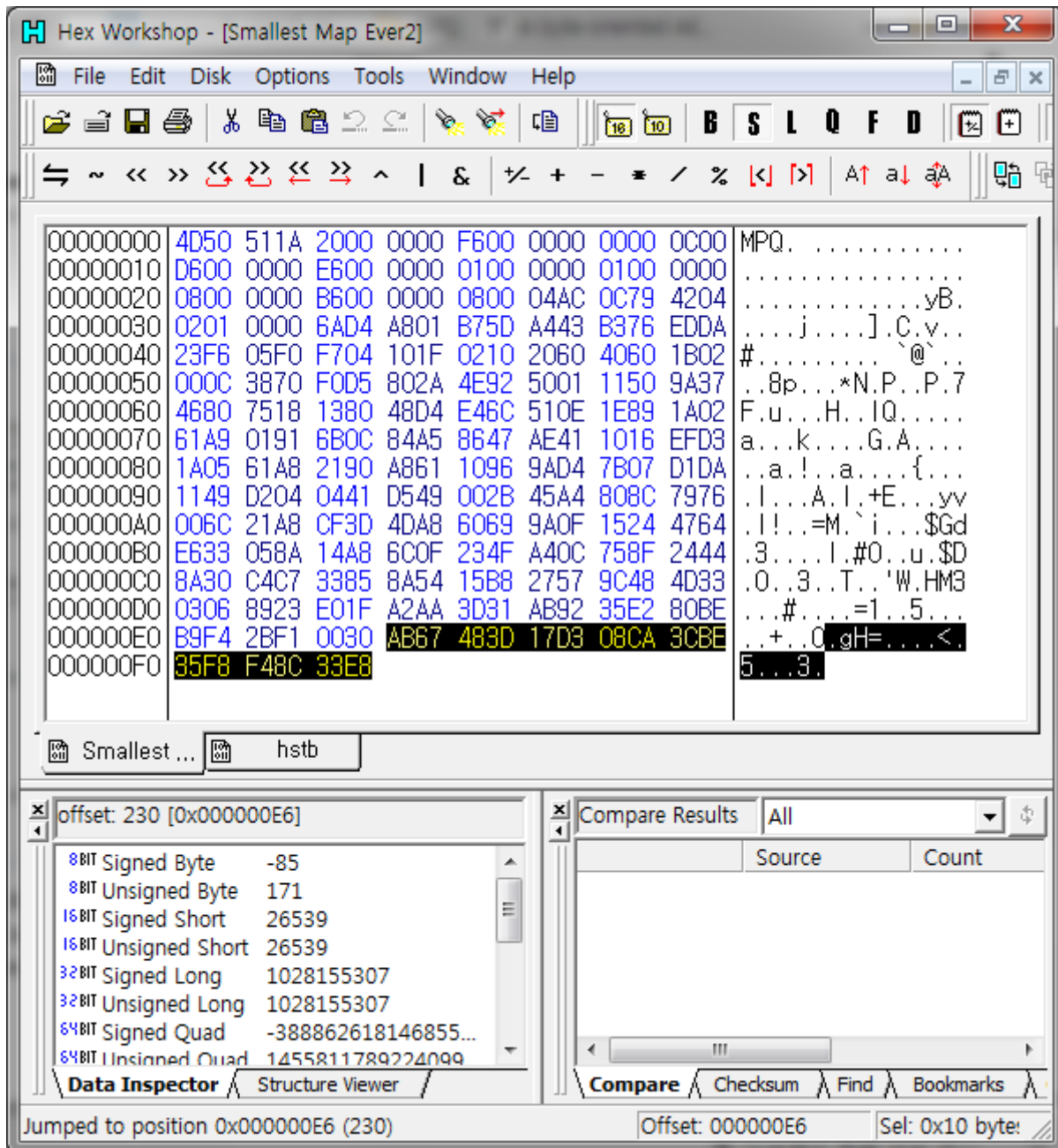
(편의상 배열은 0번째 원소부터 있다 할게요. C언어 배열과 동일하게 생각하시면 되요.)

위 표와 같이 정리할 수 있습니다.

whyask37's blog

- 언어 : 한글의 경우 1042이고 등등 그렇게 볼수도 있는데 기본값은 0. Windows의 LANGID값을 따릅니다.
- 플랫폼 : 진짜로 묻지도 따지지도 않고 0
- BET 인덱스 : BET의 0번째 엔트리에 파일 데이터 정보가 있다 는 뜻입니다.

2) BET 분석



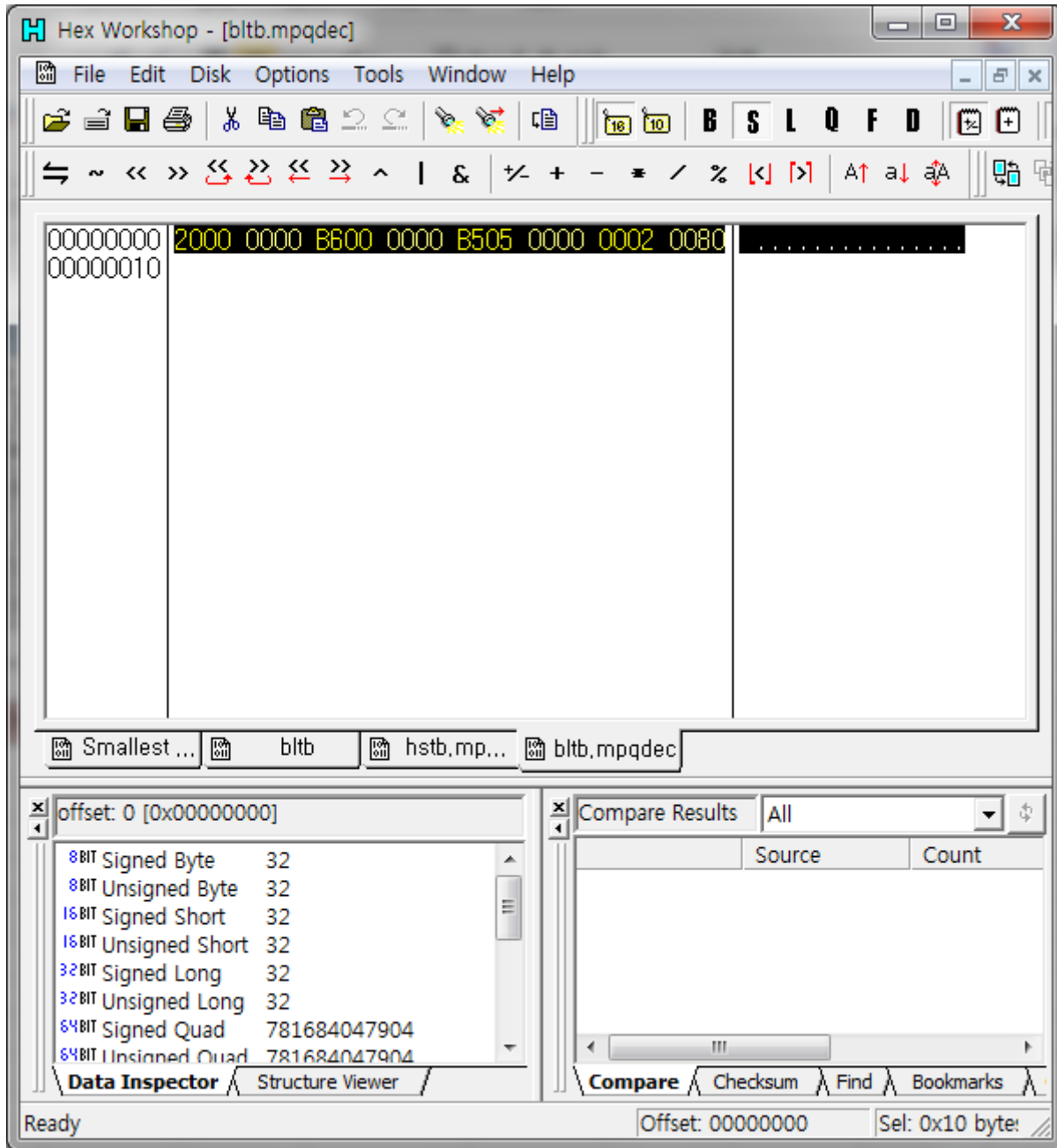
여기예요. 230byte부터 16(=1*16)byte만큼.

BET에는 다음과 같은 정보가 있어요.

- 파일 오프셋 : 파일이 어느 위치에 있는가
- 파일 크기 : 파일이 실제로 어느정도 크기인가

whyask37's blog

복호화 과정은 생략하겠습니다. "(block table)"의 MPQ_HASH_FILE_KEY값으로 복호화하시면 되요.



2000 0000 B600 0000 B505 0000 0002 0080

BET 정리			
엔트리 #0	fileOffset	32	20000000
	압축된 크기	182	B6000000
	원래 크기	1461	B5050000
	플래그	압축&파일	02000080
	총	16byte	
총 엔트리	1개	총 크기	16byte

엔트리 0에 해당하는 파일이 mpq 시작부터 32byte째부터 182byte만큼 크기를 차지하고 있으며, 압축을 풀면 1461byte가 됩니다.

whyask37's blog

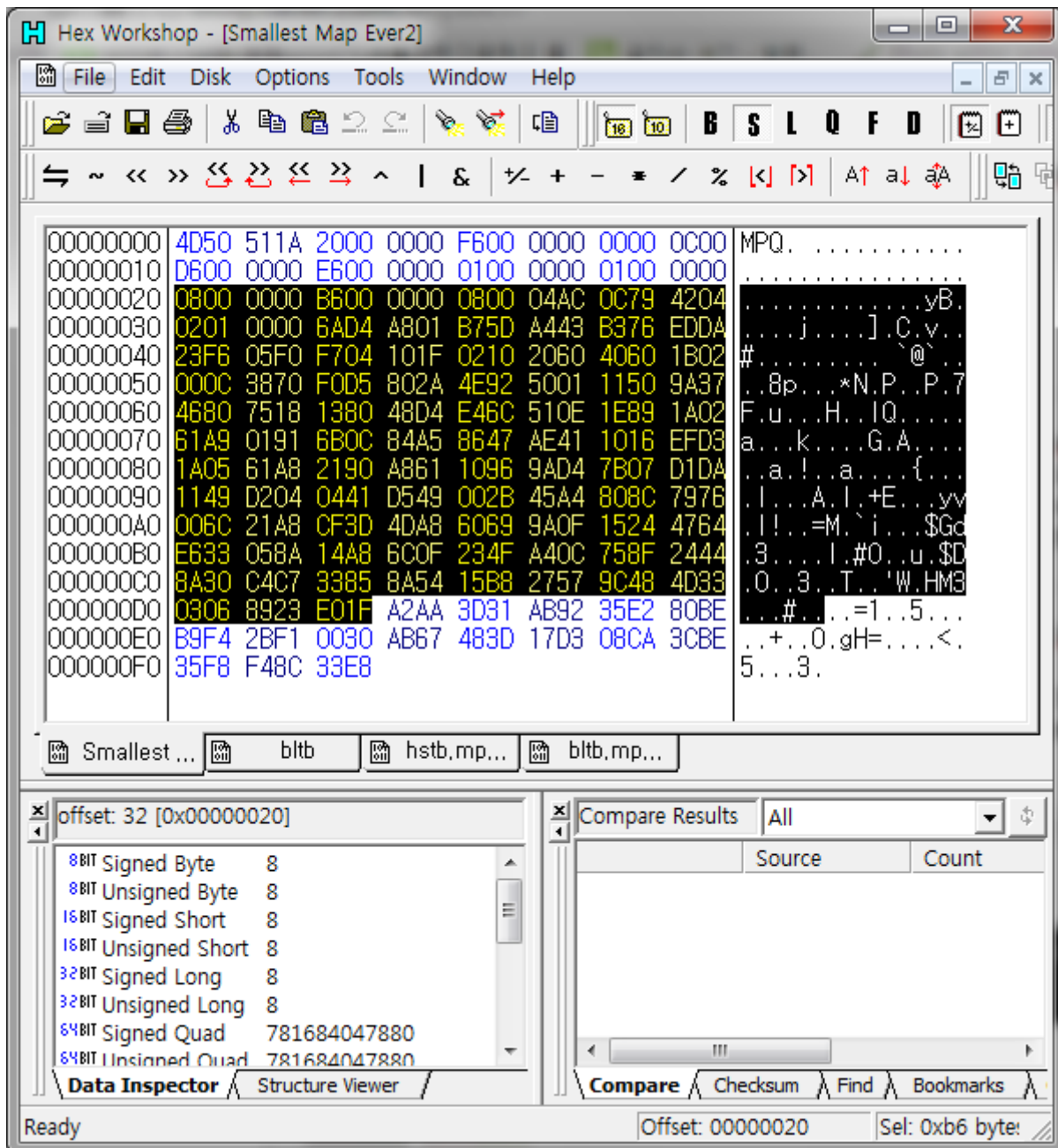
3) scenario.chk를 추출해보자.

scenario.chk의 전체 경로는 staredit\scenario.chk입니다.

파일 검색법

1. 파일 이름에 해당하는 hashA, hashB를 구합니다. (우리는 mpqhelper를 씁니다)
2. HET에서 hashA, hashB가 둘 다 일치하는 엔트리를 찾습니다. 정확히는
 - 선형 해시 테이블인데, HET의 엔트리를 찾는 해시는 MPQ_HASH_TABLE_OFFSET입니다. 여기에서 1씩 늘려 갑니다.
 - 빈 엔트리는 BET 인덱스가 0xFFFFFFFF, 삭제된 엔트리(원래 데이터가 있었음)은 BET 인덱스가 0xFFFFFFFFE입니다.
3. BET에서 해당하는 엔트리를 찾아 블록 위치, 크기, 압축을 풀었을 때 크기를 찾습니다.
4. 파일 데이터의 암호화, 압축을 풉니다.

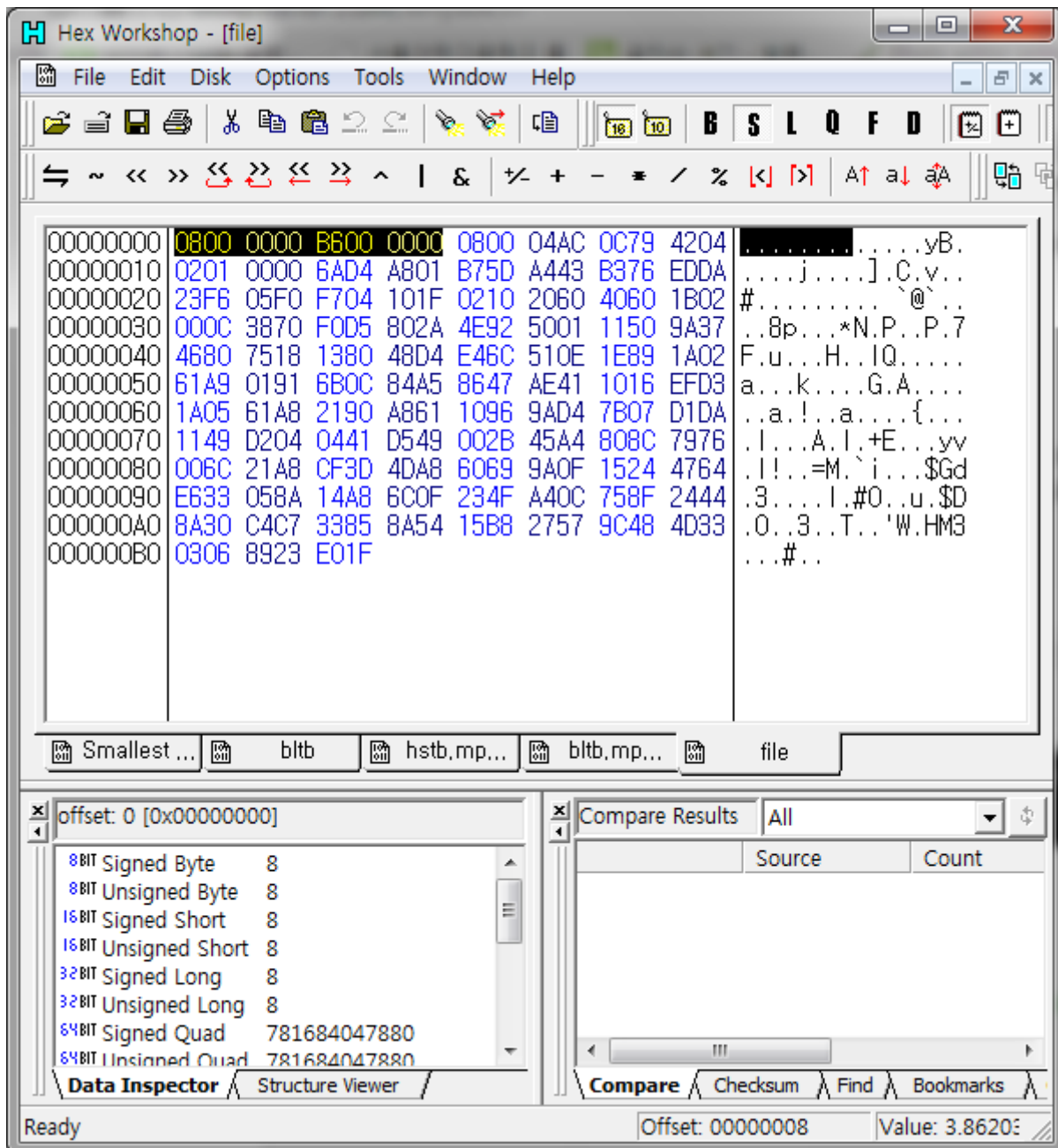
1. "staredit\scenario.chk"의 hashA = 6E6501B7, hashB = ED1EBFBC입니다... (MPQ_HASH_NAME_A, MPQ_HASH_NAME_B)
2. 어디선가 봤죠? 위에 **HET의 0번 엔트리**에 해당하네요.
3. HET의 0번 엔트리에서 **BET의 0번 엔트리**로 가라 합니다. 갔어요.
이제 32byte부터 182byte를 적당히 어찌어찌 해서 잘라서 추출해둡시다.



이제 이 파일 데이터를 압축을 풀면 됩니다. 거의 다왔어요!

sectorSizeShift에 따라서, 파일은 $1 \ll (3 + 0xC) = 32768$ 의 크기로 잘라지게 됩니다... 왜이리 큰지 잘 이해는 안가지만 뭐 그렇대요.
원래 파일의 크기는 1461byte였단니까 별 상관은 없지요. 파일은 하나의 Chunk로 잘라지게 됩니다.
(이러한 각각의 잘라진 덩어리들을 Chunk라고 부르겠습니다)

이제 파일은 아래 그림에서 블록친 부분과 나머지로 구분됩니다.



블록진 부분은 파일의 어느 부분이 어느 위치에 있냐고, 그 뒤에 부분은 파일 데이터입니다.

SectorOffsetTable

* 주의 : XXXXXXXX 꼴의 숫자는 모두 16진수 Little Endian으로 표기된 4byte 숫자입니다.

- 총 Chunk 갯수는 1개였습니다. [파일 데이터 맨 앞부분에 (Chunk갯수 + 1) * 4byte]는 SectorOffsetTable로, 다음과 같이 되어있죠?

08000000 B6000000

- 해석하자면

0번째 Chunk는 08000000 ~ B6000000 에 있다

정도의 의미가 되겠습니다.

- SectorOffsetTable이 만약 다음과 같이 구성되어있다 하면 (예시로 마음대로 만들어본 데이터입니다)

30000000 12610000 37840000 23470100 24610200 33810200 98270400

whyask37's blog

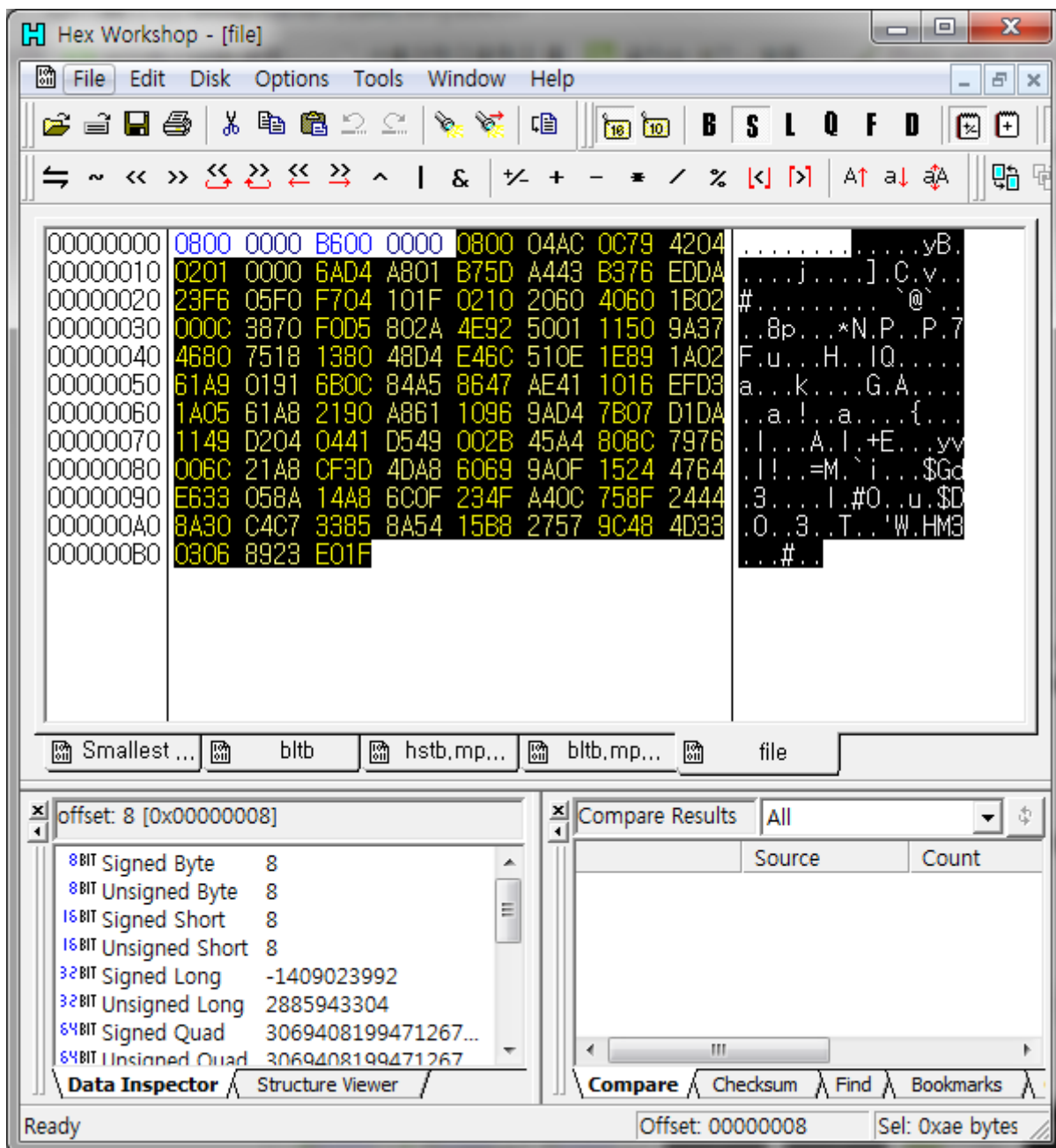
2번째 Chunk는 37840000 ~ 23470100 에 있다.
 3번째 Chunk는 23470100 ~ 24610200 에 있다.
 4번째 Chunk는 24610200 ~ 33810200 에 있다.
 5번째 Chunk는 33810200 ~ 98270400 에 있다.

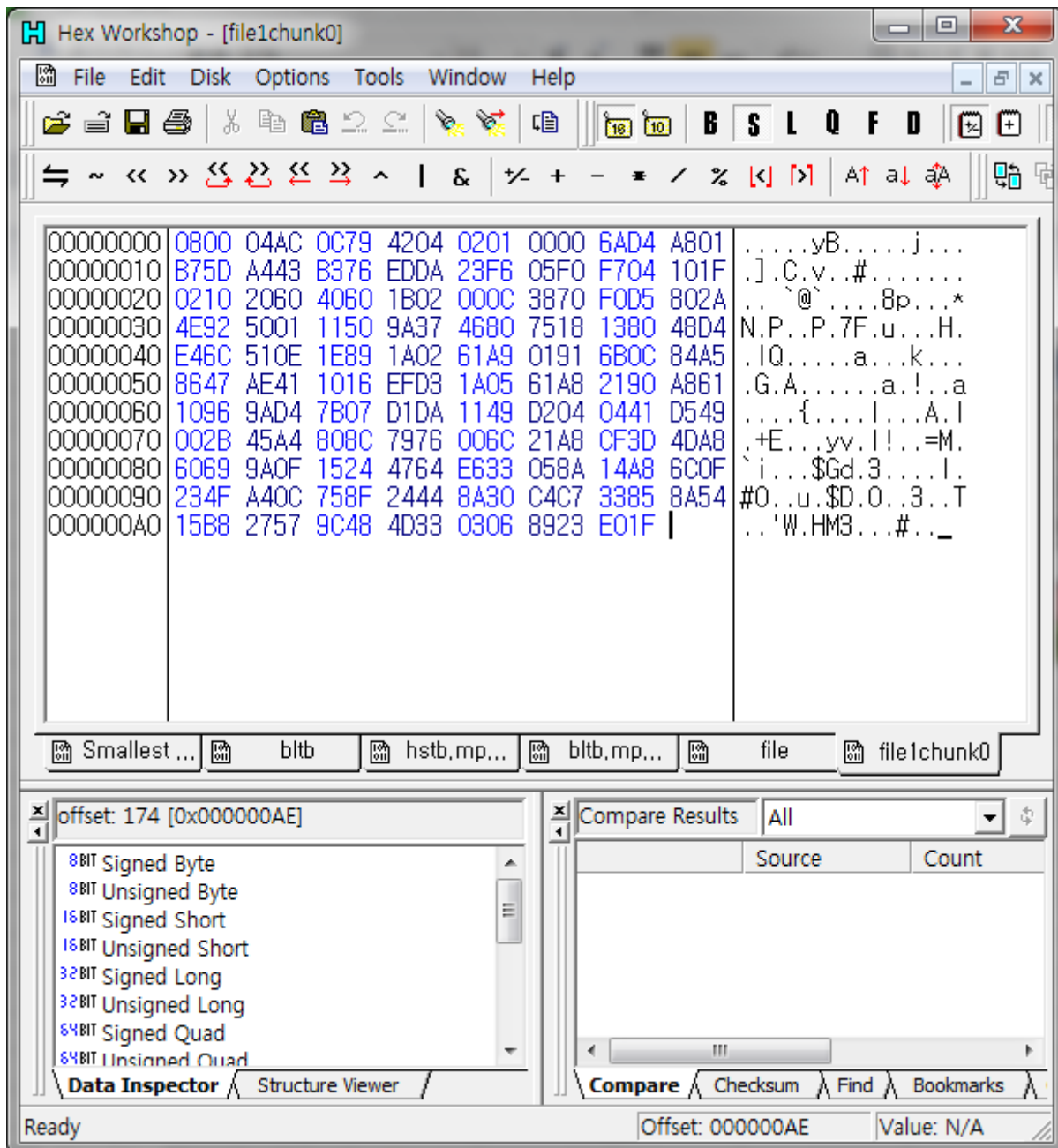
와 같이 해석됩니다.

각각의 Chunk에는 압축, 암호화 등등이 모두 따로 적용됩니다. 이로 인해서 60000번째 byte부터 읽기 위해 0~60000byte 구간을 모두 압축 해제하고 암호화를 풀고 할 필요가 없습니다. 원하는 부분만 압축과 암호화를 해제하면 되죠.

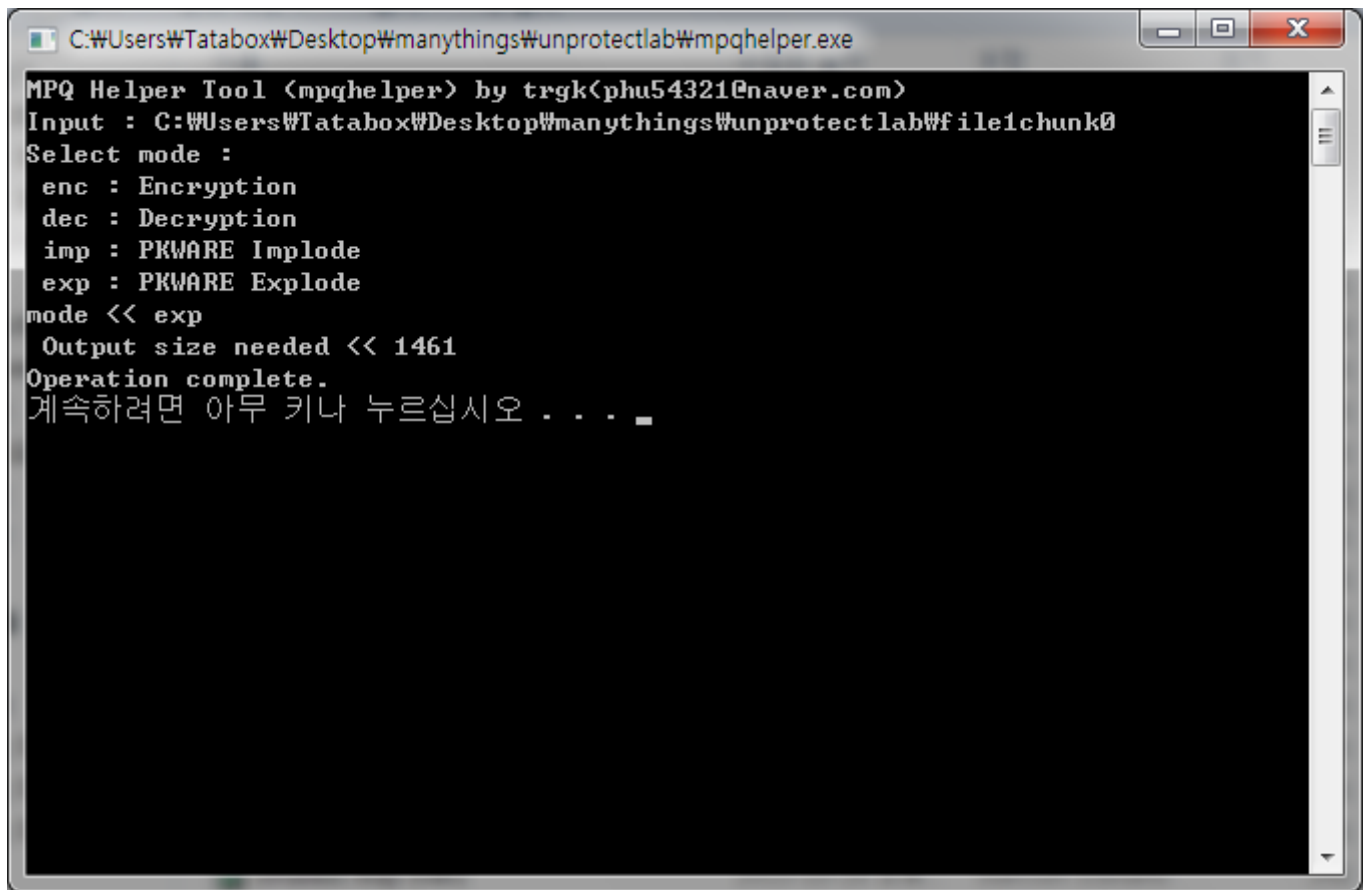
0번째 Chunk를 압축해제해봅시다.

8byte(08000000) ~ 182byte(B6000000) 영역을 적당히 file1chunk0라는 이름으로 저장해둡시다.





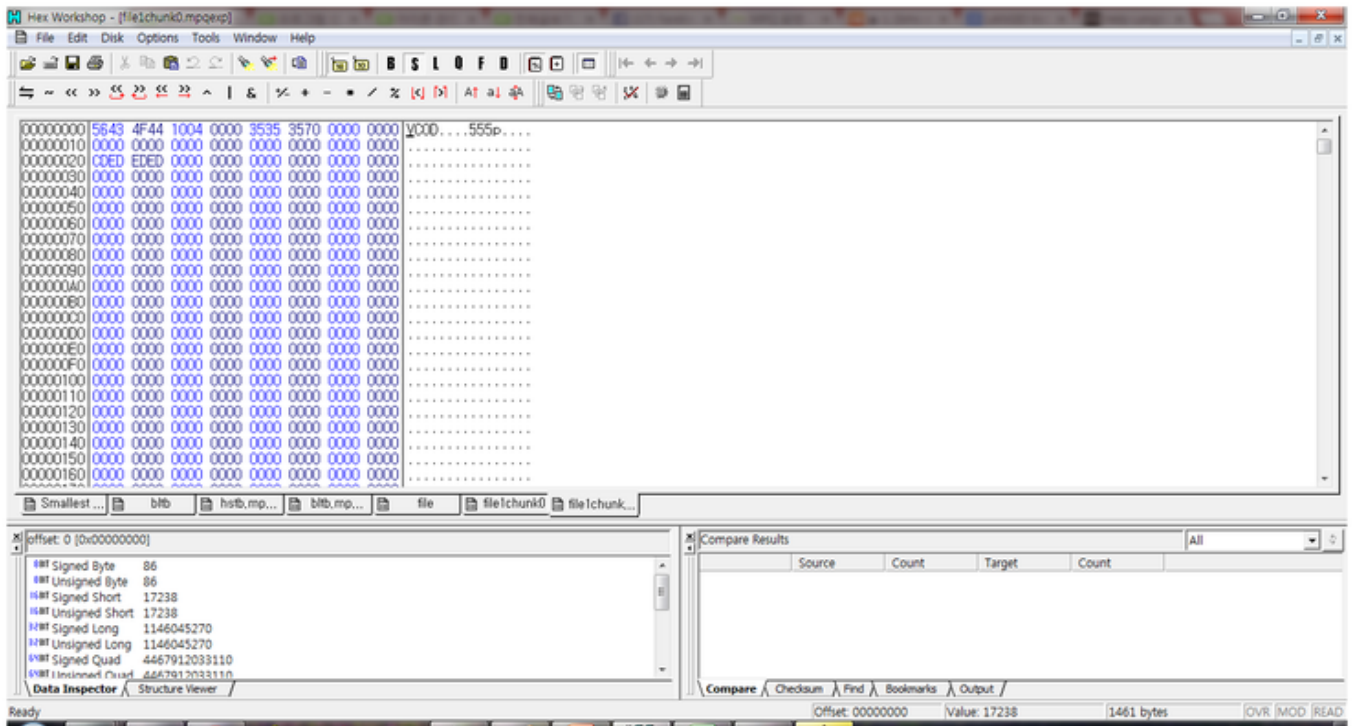
이제 이거를 해독하기 위해서 또 난관을 거쳐야 하는데, 자세한건 일단은 생략하고 간단하게 갑시다.
맨 앞에 08을 제거하고 mpqhelper의 exp(압축해제) 모드로 압축을 해제해줍니다.
 압축을 풀었을 때 크기(Output size)는 1461로 해주시고요(BET에 있었죠?)



```
C:\Users\Tatabox\Desktop\manythings\unprotectlab\mpqhelper.exe
MPQ Helper Tool <mpqhelper> by trgk<phu54321@naver.com>
Input : C:\Users\Tatabox\Desktop\manythings\unprotectlab\file1chunk0
Select mode :
  enc : Encryption
  dec : Decryption
  imp : PKWARE Implode
  exp : PKWARE Explode
mode << exp
Output size needed << 1461
Operation complete.
계속하려면 아무 키나 누르십시오 . . .
```

이렇게 하면 드디어 기다리고 기다리던 chk 파일이 추출되었습니다!

MPQ 파일 구조가 상대적으로 단순한 맵을 골라서 이렇게 쉽게 뚫습니다.
다음에는 좀 진짜 '맵'이라고 할만한 맵을 하나 골라서 조작해보겠습니다.



(음... 뭐 저래 000000이 많냐고요? 이 맵 자체가 원래 좀 그래요.)
(scenario.chk는 대충 이렇게 생겨먹었습니다.)

이야 만세

#IT·컴퓨터

첨부파일

Smallest Map Ever2.scm

mpqhelper.exe

mpqhelper.zip



왜물어

whyask37님의 블로그입니다.

이웃추가

this blog **Starpletech Unfle** Category article

My personal opinion on Unprotect...

2013. 10. 26.

4

Playing with MPQ (1) - Simple MPQ file analysis

2013. 10. 19.

11

이 블로그 인기글

5. SFmpq (ShadowFlare's MPQ Library) 와 예제

2013. 9. 11.

1

[별강의] 13. 트리거 프로그래밍 - TRIG-MRGN 루프

2014. 2. 24.

0

4. scenario.chk

2013. 9. 10.

0

[별강의] 2. 데스 사이의 대입, 더하기

2014. 1. 19.

1

whyask37's blog

[별강의] 5. 포인터 예제 - 유닛 제한 줄이기

2014. 1. 23.

0



back to top

blog market
농민후계자가 추천하는 메뚜기쌀

View in PC version