Corrupted Disk Image

Description

> 디스크 이미지가 열리지 않습니다...!
> 주어진 디스크 이미지를 복원하여 플래그를 구해주세요.
>
> FLAG: DH{something}
> something의 길이는 32자입니다.

사용한 도구

FTK Imager, HxD, HashCalc

Background

NTFS 파일 시스템의 VBR 시그니처 위치

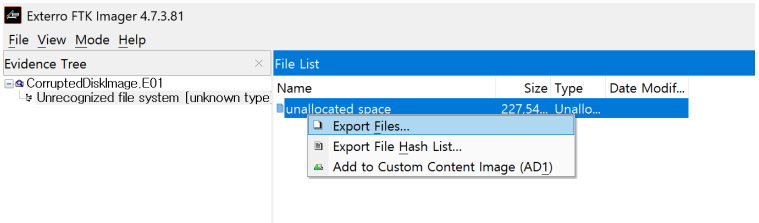
: 0xD8FFe00

NTFS 파일 시스템은 VBR의 복사본을 볼륨 끝에 저장

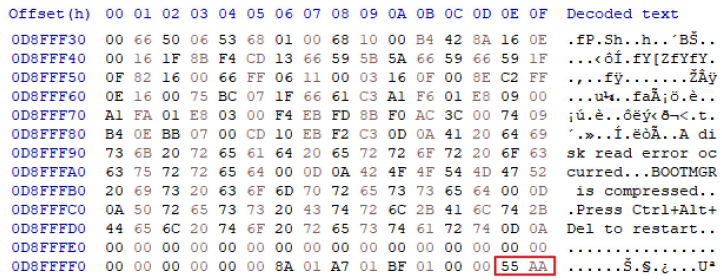
→ 맨 끝 부분에 복구용 VBR 존재

1. HxD 에서 분석하기위해 FTK Imager에서 추출

- FTK Imager만으로도 볼 수 있지만 좀 더 편리하게(크게)보기 위해 HxD에서 열어주었다.



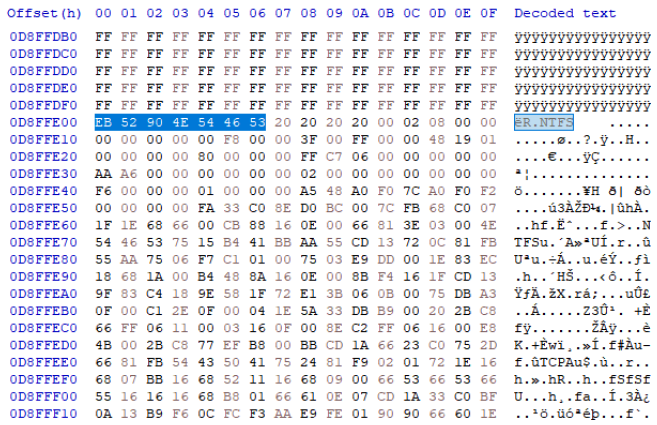
2. HxD에서 분석



Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
0D8FFF30	00	66	50	06	53	68	01	00	68	10	00	B4	42	8A	16	0E	.fP.Sh..h..´BŠ..
0D8FFF40	00	16	1F	8B	F4	CD	13	66	59	5B	5A	66	59	66	59	1F	...‹ôÍ.fY[ZfYfY.
0D8FFF50	0F	82	16	00	66	FF	06	11	00	03	16	0F	00	8E	C2	FF	.‚..fÿ.......ŽÂÿ
0D8FFF60	0E	16	00	75	BC	07	1F	66	61	C3	A1	F6	01	E8	09	00	...u¼..faÃ¡ö.è..
0D8FFF70	A1	FA	01	E8	03	00	F4	EB	FD	8B	F0	AC	3C	00	74	09	¡ú.è..ôëý‹ð¬<.t.
0D8FFF80	B4	0E	BB	07	00	CD	10	EB	F2	C3	0D	0A	41	20	64	69	´.»..Í.ëòÃ..A di
0D8FFF90	73	6B	20	72	65	61	64	20	65	72	72	6F	72	20	6F	63	sk read error oc
0D8FFFA0	63	75	72	72	65	64	00	0D	0A	42	4F	4F	54	4D	47	52	curred...BOOTMGR
0D8FFFB0	20	69	73	20	63	6F	6D	70	72	65	73	73	65	64	00	0D	is compressed..
0D8FFFC0	0A	50	72	65	73	73	20	43	74	72	6C	2B	41	6C	74	2B	.Press Ctrl+Alt+
0D8FFFD0	44	65	6C	20	74	6F	20	72	65	73	74	61	72	74	0D	0A	Del to restart..
0D8FFFE0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
0D8FFFF0	00	00	00	00	00	00	8A	01	A7	01	BF	01	00	00	55	AA	......Š.§.¿...Uª

마지막 두 바이트가 0x55AA → NTFS 파일 시스템임을 알 수 있다.

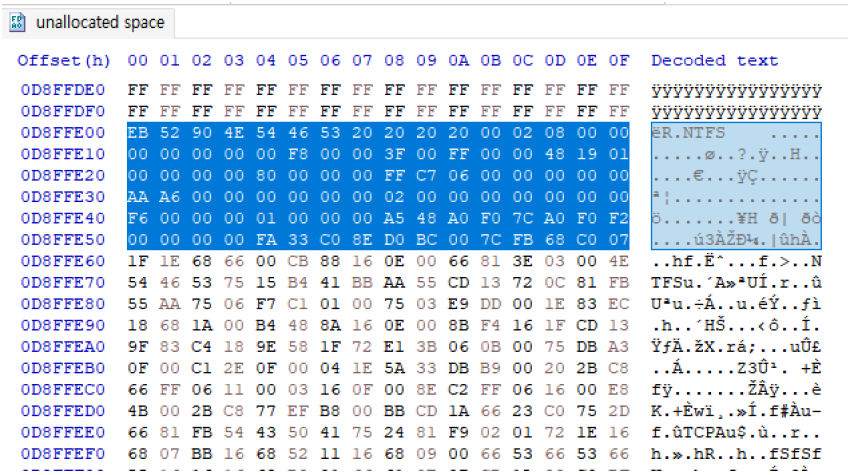
NTFS의 VBR 시그니처 offset : 0xD8FFe00



Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
0D8FFDB0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
0D8FFDC0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
0D8FFDD0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
0D8FFDE0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
0D8FFDF0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
0D8FFE00	EB	52	90	4E	54	46	53	20	20	20	00	02	08	00	00		ëR.NTFS .....
0D8FFE10	00	00	00	00	00	F8	00	00	3F	00	FF	00	00	48	19	01	.....ø..?.ÿ..H..
0D8FFE20	00	00	00	00	80	00	00	00	FF	C7	06	00	00	00	00	00	....€...ÿÇ.....
0D8FFE30	AA	A6	00	00	00	00	00	00	02	00	00	00	00	00	00	00	ª¦..............
0D8FFE40	F6	00	00	00	01	00	00	00	A5	48	A0	F0	7C	A0	F0	F2	ö.......¥H ð| ðò
0D8FFE50	00	00	00	00	FA	33	C0	8E	D0	BC	00	7C	FB	68	C0	07	....ú3ÀŽÐ¼.|ûhÀ.
0D8FFE60	1F	1E	68	66	00	CB	88	16	0E	00	66	81	3E	03	00	4E	..hf.Ë^...f.>..N
0D8FFE70	54	46	53	75	15	B4	41	BB	AA	55	CD	13	72	0C	81	FB	TFSu.´A»ªUÍ.r..û
0D8FFE80	55	AA	75	06	F7	C1	01	00	75	03	E9	DD	00	1E	83	EC	Uªu.÷Á..u.éÝ..ƒì
0D8FFE90	18	68	1A	00	B4	48	8A	16	0E	00	8B	F4	16	1F	CD	13	.h..´HŠ...‹ô..Í.
0D8FFEA0	9F	83	C4	18	9E	58	1F	72	E1	3B	06	0B	00	75	DB	A3	ŸƒÄ.žX.rá;...uÛ£
0D8FFEB0	0F	00	C1	2E	0F	00	04	1E	5A	33	DB	B9	00	20	2B	C8	..Á.....Z3Û¹. +È
0D8FFEC0	66	FF	06	11	00	03	16	0F	00	8E	C2	FF	06	16	00	E8	fÿ.......ŽÂÿ...è
0D8FFED0	4B	00	2B	C8	77	EF	B8	00	BB	CD	1A	66	23	C0	75	2D	K.+Èwï¸.»Í.f#Àu-
0D8FFEE0	66	81	FB	54	43	50	41	75	24	81	F9	02	01	72	1E	16	f.ûTCPAu$.ù..r..
0D8FFEF0	68	07	BB	16	68	52	11	16	68	09	00	66	53	66	53	66	h.».hR..h..fSfSf
0D8FFF00	55	16	16	16	68	B8	01	66	61	0E	07	CD	1A	33	C0	BF	U...h¸.fa..Í.3À¿
0D8FFF10	0A	13	B9	F6	0C	FC	F3	AA	E9	FE	01	90	90	66	60	1E	..¹ö.üóªéþ...f`.

0xD8FFe00 의 위치를 보면 **EB 52 90 4E 54 46 53** 확인 가능

3. 복구를 위한 덮어쓰기



Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
0D8FFDE0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
0D8FFDF0	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	ÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿÿ
0D8FFE00	EB	52	90	4E	54	46	53	20	20	20	00	02	08	00	00		ëR.NTFS .....
0D8FFE10	00	00	00	00	00	F8	00	00	3F	00	FF	00	00	48	19	01	.....ø..?.ÿ..H..
0D8FFE20	00	00	00	00	80	00	00	00	FF	C7	06	00	00	00	00	00	....€...ÿÇ.....
0D8FFE30	AA	A6	00	00	00	00	00	00	02	00	00	00	00	00	00	00	ª¦..............
0D8FFE40	F6	00	00	00	01	00	00	00	A5	48	A0	F0	7C	A0	F0	F2	ö.......¥H ð| ðò
0D8FFE50	00	00	00	00	FA	33	C0	8E	D0	BC	00	7C	FB	68	C0	07	....ú3ÀŽÐ¼.|ûhÀ.
0D8FFE60	1F	1E	68	66	00	CB	88	16	0E	00	66	81	3E	03	00	4E	..hf.Ë^...f.>..N
0D8FFE70	54	46	53	75	15	B4	41	BB	AA	55	CD	13	72	0C	81	FB	TFSu.´A»ªUÍ.r..û
0D8FFE80	55	AA	75	06	F7	C1	01	00	75	03	E9	DD	00	1E	83	EC	Uªu.÷Á..u.éÝ..ƒì
0D8FFE90	18	68	1A	00	B4	48	8A	16	0E	00	8B	F4	16	1F	CD	13	.h..´HŠ...‹ô..Í.
0D8FFEA0	9F	83	C4	18	9E	58	1F	72	E1	3B	06	0B	00	75	DB	A3	ŸƒÄ.žX.rá;...uÛ£
0D8FFEB0	0F	00	C1	2E	0F	00	04	1E	5A	33	DB	B9	00	20	2B	C8	..Á.....Z3Û¹. +È
0D8FFEC0	66	FF	06	11	00	03	16	0F	00	8E	C2	FF	06	16	00	E8	fÿ.......ŽÂÿ...è
0D8FFED0	4B	00	2B	C8	77	EF	B8	00	BB	CD	1A	66	23	C0	75	2D	K.+Èwï¸.»Í.f#Àu-
0D8FFEE0	66	81	FB	54	43	50	41	75	24	81	F9	02	01	72	1E	16	f.ûTCPAu$.ù..r..
0D8FFEF0	68	07	BB	16	68	52	11	16	68	09	00	66	53	66	53	66	h.».hR..h..fSfSf

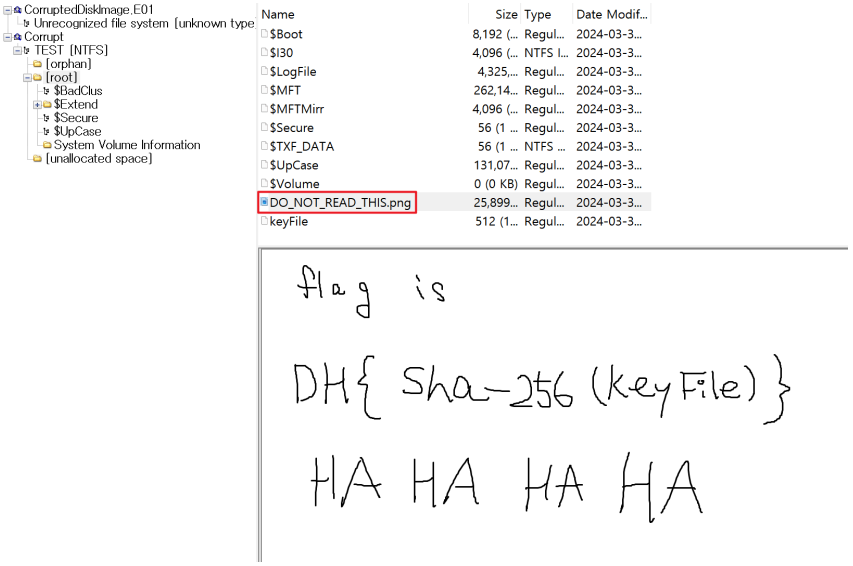
파일을 복구하기 위해 복구용 VBR 부분 (시그니처 부분 이후) 을 맨 앞에 덮어쓴다.

이 때 주의할 점은 붙여넣기가 아닌 덮어쓰기를 해야한다(ctrl+B)

unallocated space

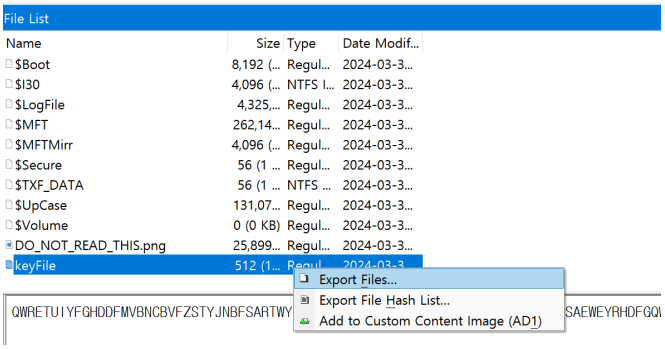
Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
00000000	EB	52	90	4E	54	46	53	20	20	20	20	00	02	08	00	00	ëR.NTFS .....
00000010	00	00	00	00	00	F8	00	00	3F	00	FF	00	00	48	19	01	.....ø..?.ÿ..H..
00000020	00	00	00	00	80	00	00	00	FF	C7	06	00	00	00	00	00	....€...ÿÇ.....
00000030	AA	A6	00	00	00	00	00	00	02	00	00	00	00	00	00	00	ª¦..............
00000040	F6	00	00	00	01	00	00	00	A5	48	A0	F0	7C	A0	F0	F2	ö.......¥H ð| ðò
00000050	00	00	00	00	FA	33	C0	8E	D0	BC	00	7C	FB	68	C0	07	....ú3ÀŽÐ¼.|ûhÀ.
00000060	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	ªªªªªªªªªªªªªªªª
00000070	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	ªªªªªªªªªªªªªªªª
00000080	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	ªªªªªªªªªªªªªªªª
00000090	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	ªªªªªªªªªªªªªªªª
000000A0	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	AA	ªªªªªªªªªªªªªªªª

4. FTK Imager에서 복구 확인



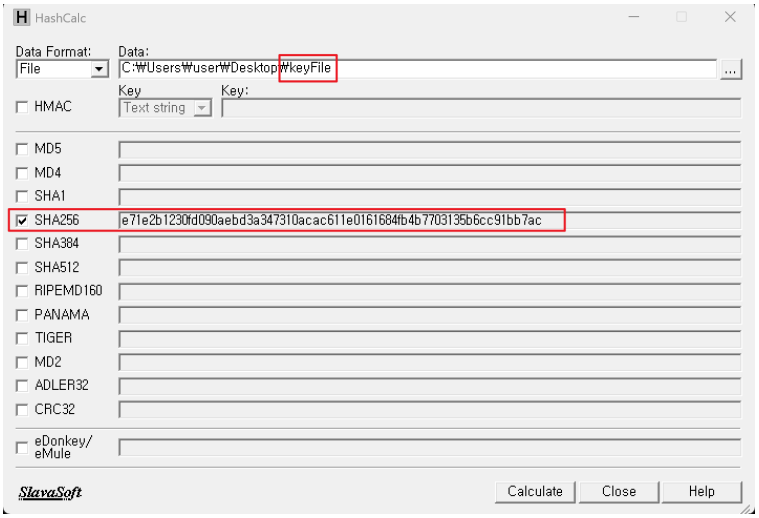
저장한 후 FTK Imager에서 열어보니 flag의 힌트를 찾을 수 있었다.

5. keyFile 값 sha-256 변환



hash값을 알기위해 HashCalc라는 도구를 사용해주었다.

FTK Imager로 추출 후 HashCalc로 열어 계산해주면 SHA256값을 알 수 있다.



FLAG

DH{e71e2b1230fd090aebd3a347310acac611e0161684fb4b7703135b6cc91bb7ac}