<Txt 파일 찾기 프로그램 만들기>

목표 : 가설을 토대로 Header 도 footer 도 없는 경우 (txt) : 는 삭제되었을 때 생긴 앞의 [Trash Info] = 5b54 7261 7368 2049 6e66 6f5d 0a50 를 인식하고 복원하는 프로그램 만들기

환경: img 파일에는 삭제된 txt 파일 1개, 살아있는 txt 파일 1개가 존재

1. 위의 가설이 맞는지 확인

```
by@ubuntu:~/dump$ sigfind -b 4096 5b5472 03_dump_txt.img
Block size: 4096  Offset: 0  Signature: 5B5472
Block: 33017 (-)
Block: 33199 (+182)
Block: 33797 (+598)
error reading bytes 507619
by@ubuntu:~/dump$
```

⇨ Sigfind 로 5b5672([Tras..]를 검색해본 결과 다음과 같은 결과가 나왔다.

⇨ 여기서 33797 은 recovertxt1, 33199는 recovertxt2, 33017 은 recovertxt3 으로 dd 명령어를 사용하여 덤프 뜬 후, 파일 구조를 확인해보겠다.

(1) Recovertxt1(33199)

```
by@ubuntu:~/dump$ dd if=03_dump_txt.img bs=4096 count=18 skip=33797 >recovertxt
1
```

```
by@ubuntu:~/dump$ xxd recovertxt1
00000000: 5b54 7261 7368 2049 6e66 6f5d 0a50 6174  [Trash Info].Pat
00000010: 683d 7468 6973 5f69 735f 7465 7374 5f64  h=this_is_test_d
00000020: 656c 6574 655f 322e 7478 740a 4465 6c65  elete_2.txt.Dele
00000030: 7469 6f6e 4461 7465 3d32 3031 392d 3130  tionDate=2019-10
00000040: 2d30 3854 3136 3a32 353a 3530 0a00 0000  -08T16:25:50....
00000050: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00000060: 0000 0000 0000 0000 0000 0000 0000 0000  ................
```

⇨ Recovertxt1 같은 경우에는 전부 앞에 txt 삭제된 정보는 존재하지만 전부 0으로 패딩된 덤프파일이 나왔다. Count 크기를 333 까지 높여 더욱 크게 떠봤음에도 불구하고 내용은 마찬가지로 0으로 패딩된 결과가 나왔다. => 실패

(2) Recovertxt2 (33199)

```
by@ubuntu:~/dump$ dd if=03_dump_txt.img bs=4096 count=18 skip=33199 >recovertxt
2
```

```
by@ubuntu:~/dump$ xxd recovertxt2
00000000: 5b54 7261 7368 2049 6e66 6f5d 0a50 6174  [Trash Info].Pat
00000010: 683d 7468 6973 5f69 735f 6465 6c65 7465  h=this_is_delete
00000020: 642e 7478 740a 4465 6c65 7469 6f6e 4461  d.txt.DeletionDa
00000030: 7465 3d32 3031 392d 3130 2d30 3854 3136  te=2019-10-08T16
00000040: 3a31 313a 3131 0a00 0000 0000 0000 0000  :11:11..........
00000050: 0000 0000 0000 0000 0000 0000 0000 0000  ................
```

⇨ Recovertxt2 같은 경우에도 앞의 삭제 정보가 나왔다

⇨ 하지만 이번 경우에는 삭제 정보뒤 0 패딩 이후로 아래와 같은 txt 본문이 나왔다

```
00000fd0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00000fe0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00000ff0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00001000: 7468 6973 2069 7320 7465 7374 2074 6869  this is test thi
00001010: 7320 6973 2074 6573 7420 7468 6973 2069  s is test this i
00001020: 7320 7465 7374 2074 6869 7320 6973 2074  s test this is t
00001030: 6573 7420 7468 6973 2069 7320 7465 7374  est this is test
00001040: 2074 6869 7320 6973 2074 6573 7420 7468   this is test th
00001050: 6973 2069 7320 7465 7374 2074 6869 7320  is is test this
00001060: 6973 2074 6573 7420 7468 6973 2069 7320  is test this is
00001070: 7465 7374 2074 6869 7320 6973 2074 6573  test this is tes
```

```
00011f90: 6573 7420 7468 6973 2069 7320 7465 7374  est this is test
00011fa0: 2074 6869 7320 6973 2074 6573 7420 7468   this is test th
00011fb0: 6973 2069 7320 7465 7374 2074 6869 7320  is is test this
00011fc0: 6973 2074 6573 7420 7468 6973 2069 7320  is test this is
00011fd0: 7465 7374 2074 6869 7320 6973 2074 6573  test this is tes
00011fe0: 7420 7468 6973 2069 7320 7465 7374 2074  t this is test t
00011ff0: 6869 7320 6973 2074 6573 7420 7468 6973  his is test this
by@ubuntu:~/dump$
```

⇨ 보면 끝이 없이 계속된다. Dd 명령어의 count 크기를 높여 보겠다.

```
00050fe0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00050ff0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00051000: 7468 6973 5f69 735f 6465 6c65 7465 645f  this_is_deleted_
00051010: 7478 745f 7468 6973 5f69 735f 6465 6c65  txt_this_is_dele
00051020: 7465 645f 7478 745f 7468 6973 5f69 735f  ted_txt_this_is_
00051030: 6465 6c65 7465 645f 7478 745f 7468 6973  deleted_txt_this
00051040: 5f69 735f 6465 6c65 7465 645f 7478 745f  _is_deleted_txt_
00051050: 7468 6973 5f69 735f 6465 6c65 7465 645f  this_is_deleted_
00051060: 7478 745f 7468 6973 5f69 735f 6465 6c65  txt_this_is_dele
00051070: 7465 645f 7478 745f 7468 6973 5f69 735f  ted_txt_this_is_
00051080: 6465 6c65 7465 645f 7478 745f 7468 6973  deleted_txt_this
00051090: 5f69 735f 6465 6c65 7465 645f 7478 745f  _is_deleted_txt_
000510a0: 7468 6973 5f69 735f 6465 6c65 7465 645f  this_is_deleted_
000510b0: 7478 745f 7468 6973 5f69 735f 6465 6c65  txt_this_is_dele
000510c0: 7465 645f 7478 745f 7468 6973 5f69 735f  ted_txt_this_is_
000510d0: 6465 6c65 7465 645f 7478 745f 7468 6973  deleted_txt_this
```

⇨ 실종되었던 삭제 파일 찾음..

⇨ Recovertxt2 => 결론 : 삭제 정보로 위치를 추측할 수 없을듯 싶다.

(3) recovertxt3(33017)

```
by@ubuntu:~/dump$ dd if=03_dump_txt.img bs=4096 count=18 skip=33017 >recovertxt3
18+0 records in
18+0 records out
73728 bytes (74 kB, 72 KiB) copied, 0.00112873 s, 65.3 MB/s
```

⇨ 이것은 txt 정보가 아니고 jpg 삭제 정보이다.

```
by@ubuntu:~/dump$ xxd recovertxt3
00000000: 5b54 7261 7368 2049 6e66 6f5d 0a50 6174  [Trash Info].Pat
00000010: 683d 6361 7425 3238 6465 6c65 7465 2532  h=cat%28delete%2
00000020: 392e 6a70 670a 4465 6c65 7469 6f6e 4461  9.jpg.DeletionDa
00000030: 7465 3d32 3031 392d 3130 2d30 3654 3039  te=2019-10-06T09
00000040: 3a35 383a 3430 0a00 0000 0000 0000 0000  :58:40..........
00000050: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00000060: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00000070: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00000080: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00000090: 0000 0000 0000 0000 0000 0000 0000 0000  ................
000000a0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
000000b0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
000000c0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
000000d0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
000000e0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
```

```
00000ff0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00001000: ffd8 ffe0 0010 4a46 4946 0001 0100 0001  ......JFIF......
00001010: 0001 0000 ffdb 0084 0009 0607 1312 1215  ................
00001020: 1212 1315 1515 1515 1515 1515 1715 1515  ................
00001030: 1515 1515 1516 1615 1515 1518 1d28 2018  .............( .
00001040: 1a25 1d15 1521 3121 2529 2b2e 2e2e 171f  .%...!1!%)+.....
00001050: 3338 332d 3728 2d2e 2b01 0a0a 0a0e 0d0e  383-7(-.+.......
00001060: 1710 101a 2d1d 1d1d 2d2d 2d2d 2d2d 2d2d  ....-...--------
00001070: 2d2d 2d2d 2d2d 2d2b 2d2d 2d2d 2d2d 2d2d  -------+--------
00001080: 2d2d 2d2d 2d2d 2d2d 2d2d 2d2d 2d2d 2d2d  ----------------
00001090: 2d2d 2d2d 2d2d 2b2d 2d2d ffc0 0011 0800  ------+---------
000010a0: ca00 f903 0122 0002 1101 0311 01ff c400  .....".........
000010b0: 1c00 0002 0301 0101 0100 0000 0000 0000  ................
000010c0: 0000 0405 0203 0607 0100 08ff c400 3c10  ..............<.
000010d0: 0001 0302 0403 0506 0405 0403 0100 0000  ................
000010e0: 0100 0203 0411 0512 2131 0641 5113 6171  ........!1.AQ.aq
000010f0: 8191 1422 32a1 b1f0 52c1 d1e1 0715 2342  ..."2...R.....#B
00001100: f133 8292 a262 83c2 72ff c400 1901 0003  .3...b..r.......
00001110: 0101 0100 0000 0000 0000 0000 0000 0001  ................
00001120: 0302 0405 ffc4 0024 1100 0202 0202 0202  .......$........
00001130: 0203 0000 0000 0000 0000 0102 1103 2112  ..............!.
00001140: 3104 4113 5122 7114 3261 ffda 000c 0301  1.A.Q"q.2a......
```

⇨ 아래가보면 JFIF 의 정보가 나온다.

⇨ 또 다른 jpeg 데이터도 2개 나온다

⇨ 그리고 나서 txt 가 나온다. => 이거는 삭제되지 않은 것이다.

```
00007fb0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00007fc0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00007fd0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00007fe0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00007ff0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00008000: 7468 6973 2069 7320 7465 7374 2030 2074  this is test 0 t
00008010: 6869 7320 6973 2074 6573 7420 3020 7468  his is test 0 th
00008020: 6973 2069 7320 7465 7374 2030 2074 6869  is is test 0 thi
00008030: 7320 6973 2074 6573 7420 3020 7468 6973  s is test 0 this
00008040: 2069 7320 7465 7374 2030 2074 6869 7320   is test 0 this
00008050: 6973 2074 6573 7420 3020 7468 6973 2069  is test 0 this i
00008060: 7320 7465 7374 2030 2074 6869 7320 6973  s test 0 this is
00008070: 2074 6573 7420 3020 7468 6973 2069 7320   test 0 this is
00008080: 7465 7374 2030 2074 6869 7320 6973 2074  test 0 this is t
00008090: 6573 7420 3020 7468 6973 2069 7320 7465  est 0 this is te
000080a0: 7374 2030 2074 6869 7320 6973 2074 6573  st 0 this is tes
```

```
00008790: 6869 7320 6973 2074 6573 7420 3020 7468  his is test 0 th
000087a0: 6973 2069 7320 7465 7374 2030 2076 0a00  is is test 0 v..
000087b0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
000087c0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
000087d0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
000087e0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
000087f0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00008800: 0000 0000 0000 0000 0000 0000 0000 0000  ................
```

⇨ 이렇게 살아 있는 txt 내용이 나오고 다스 jfif 정보가 나왔다

```
00008ff0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
00009000: ffd8 ffe0 0010 4a46 4946 0001 0100 0001  ......JFIF......
00009010: 0001 0000 ffdb 0084 0009 0607 1213 1215  ................
00009020: 1212 1215 1515 1515 1015 1515 1515 1515  ................
00009030: 1217 1515 1516 1615 1515 1518 1d28 2018  .............( .
00009040: 1a25 1b15 1521 3121 2529 2b2e 2e2e 171f  .%...!1!%)+.....
00009050: 3338 332d 3728 2d2e 2b01 0a0a 0a0e 0d0e  383-7(-.+.......
00009060: 1a10 1017 2b1d 1e1d 2d2d 2d2d 2d2d 2d2d  ....+...--------
00009070: 2d2d 2d2d 2d2d 2d2d 2d2d 2b2d 2d2d 2d2d  ----------+-----
00009080: 2d2d 2d2d 2d2d 2b2d 2d2d 2d2d 372d 2d2d  ------+-----7---
```

```
0000aff0: 0000 0000 0000 0000 0000 0000 0000 0000  ................
0000b000: ffd8 ffe0 0010 4a46 4946 0001 0100 0001  ......JFIF......
0000b010: 0001 0000 ffe1 0060 4578 6966 0000 4949  .......`Exif..II
0000b020: 2a00 0800 0000 0300 3101 0200 0700 0000  *.......1.......
0000b030: 3200 0000 3b01 0200 1000 0000 3900 0000  2...;.......9...
0000b040: 9882 0200 0f00 0000 4900 0000 0000 0000  ........I.......
0000b050: 476f 6f67 6c65 004e 6174 6861 6e20 5065  Google.Nathan Pe
0000b060: 7465 7273 656e 0028 4329 204e 5050 6963  tersen.(C) NPPic
0000b070: 7475 7265 7300 ffe2 0bf8 4943 435f 5052  tures.....ICC_PR
0000b080: 4f46 494c 4500 0101 0000 0be8 0000 0000  OFILE...........
0000b090: 0200 0000 6d6e 7472 5247 4220 5859 5a20  ....mntrRGB XYZ
```

⇨ 이 파일이 큰지 끝이 없다.

최종 결론 : trashinfo 의 시그니처로 삭제된 파일이 무엇인지는 알 수 있지만 삭제된 경로는 알 수 없다.