1. 개요

삭제된 파일의 inode에 data block에 대한 정보가 남아있지 않다.

우리는 inode를 이용한 삭제된 파일의 복구를 시도할 수 없게 되었다.

1. 방법

파일 signature를 이용해 파일의 시작점을 찾을 수 있다는 것을 이용하여 삭제된 파일을 복구한다.

1. 본문

다음 장부터는 본문의 내용이 이어진다.

1). fsstat [filename]

|  |
| --- |
| noname@ubuntu:~$ fsstat dump\_new.img  …  CONTENT INFORMATION  --------------------------------------------  Block Groups Per Flex Group: 16  Block Range: 0 - 507618  Block Size: 4096  Free Blocks: 489959  … |

2) sigfind –b [block\_size] [hex\_signature] [filename]

|  |
| --- |
| noname@ubuntu:~$ sigfind -b 4096 FFD8 dump\_new.img  Block size: 4096 Offset: 0 Signature: FFD8  Block: 33018 (-)  Block: 33020 (+2)  Block: 33022 (+2)  Block: 33026 (+4)  Block: 33028 (+2)  Block: 33124 (+96)  error reading bytes 507619  noname@ubuntu:~$ |
| \*\*\* 참고 시그니처 \*\*\*  JPG : FFD8 |

3) 파일 복구 (Example 1)

|  |
| --- |
| noname@ubuntu:~$ dd if=dump\_new.img bs=4096 count=9 skip=33124 > recover  9+0 records in  9+0 records out  36864 bytes (37 kB, 36 KiB) copied, 0.000264919 s, 139 MB/s  noname@ubuntu:~$ |



<복구한 파일>

4) jpg로 확장자 정해줌

|  |
| --- |
| noname@ubuntu:~$ cat recover >recover.jpeg  noname@ubuntu:~$ ls  Desktop Documents Downloads dump\_new.img Music Pictures Public recover recover.jpeg Templates Videos  noname@ubuntu:~$ |

\*\* 파일 복구 (Example 2)

|  |
| --- |
| noname@ubuntu:~$ dd if=dump\_new.img bs=4096 count=2 skip=33026 > recover  4+0 records in  4+0 records out  16384 bytes (16 kB, 16 KiB) copied, 0.00832166 s, 2.0 MB/s  noname@ubuntu:~$ |



<복구한 파일>

1. 결론

Sigfind는 **시작 데이터 블록이 몇 번째 데이터 블록**인지와 그 앞 데이터 블록에 저장된 파일의 시작부분과의 거리를 알려준다.

파일A의 데이터블록과 파일B의 데이터블록 사이가 0으로 채워져 있다면 그 차이만큼 복구하면 될 일이다.

하지만 만약 디스크에 쓰기와 지우기를 자주 해서 그 중간에 쓰레기 값이 남아있다면 쓰레기 값이 함께 복구될 수 있다.

* 파일 A 뒤의 파일 B도 항상 같은 signature일리가 없다.

따라서 파일 A와 파일 B의 데이터 블록의 거리를 알 수 없다.

그러나 그림파일의 경우 데이터의 뒤가 쓰레기 값이어도 원본 그림을 알아보는 데는 지장이 없다.

* Questions

1 파일 A의 데이터 블록이 끝나는 구간을 어떻게 알아낼 것인가?

2. 그림파일이 아니고 다른 파일이라면 어떻게 될까?