$MFT, $USNJRNL, $LOGFILE에서 획득할 수 있는 침해사고 관련 정보

김 태 훈
Email: kimfrancesco@gmail.com
Blog: kkomak.wordpress.com

침해사고 조사 시, 분석 필요한 MFT, USNJRNL, LOGFILE 관련항목

- 파일 시간 조작 확인
- 특정 시간에 생성 및 삭제된 파일 정보
- 파일 이름 및 사이즈 정보
- $DATA영역에 존재하는 데이터

파일 시간 조작 확인

- 악성코드는 파일시스템에서 보여지는 시간 값을 조작하는 경우가 있으며, 이는 침해사고 조사 시의 타임라인 분석에서 은닉되도록 하기 위한 기법.

- 윈도우 터널링 기법인 아닌, 일반적인 파일 시간 조작 프로그램을 가능하여 시간을 조작하게 되면, 그 조작된 데이터는 $STD_INFO에 저장된 시간 값이 변경이 발생되고, $FILE_NAME의 시간 값은 변경되지 않음

- 이 두 개의 필드의 파일 생성 및 수정 필드 시간 값을 대조하여, 틀린 시간 값이 있는 것에 대해서는 시간 조작을 의심해야 함.

- 단, 윈도우에서 제공하는 파일 터널링 기능을 이용할 경우, STD_INFO와 FILE_NAME 생성 시간 등을 동일하도록 만들 수 있음.

  - 이럴 경우, $USNJRNL의 파일 생성 시간 정보를 참조하여 비교해야 함.

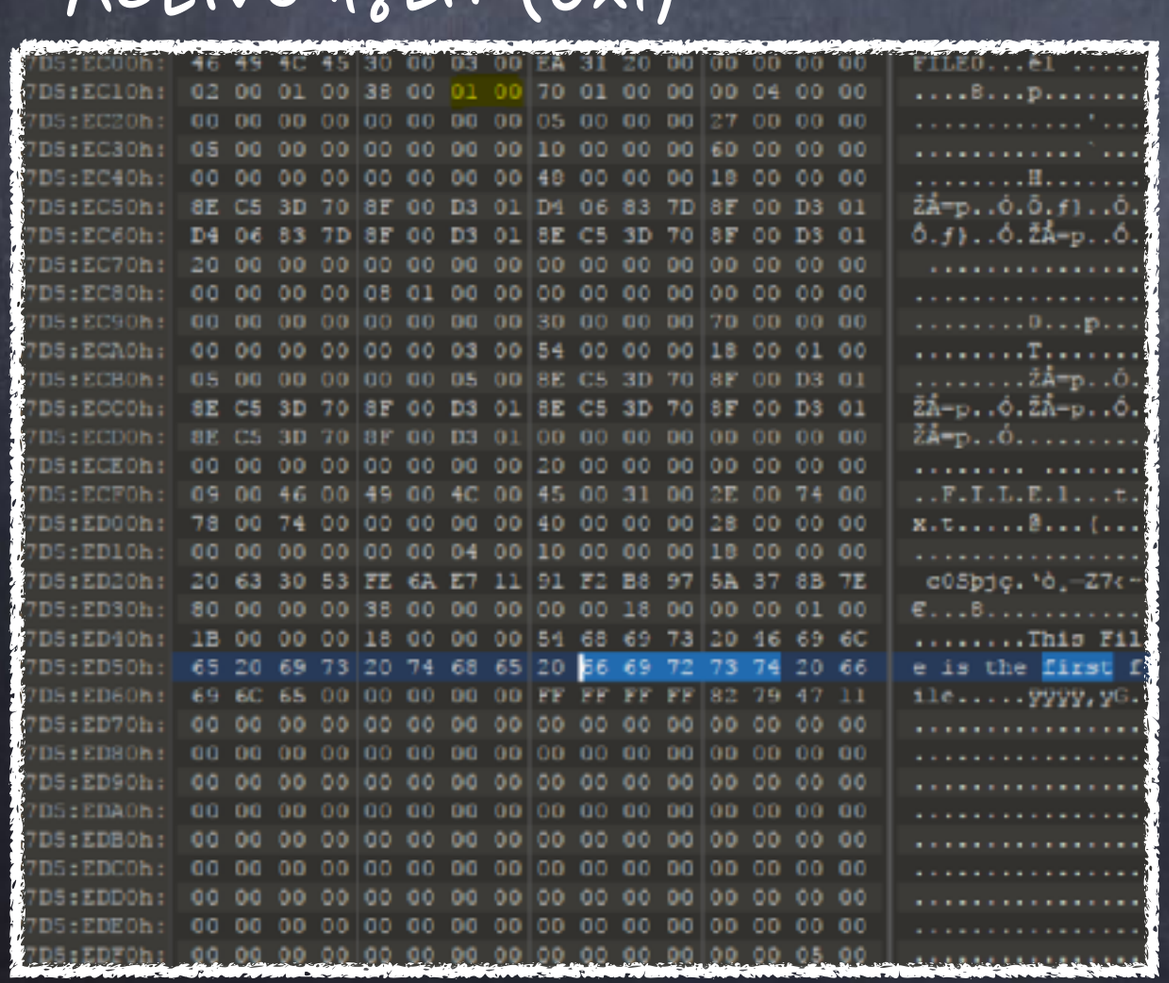
특정 시간에 생성 및 삭제된 파일 정보

- 사고 분석 과정에서 침해사고 추정 시간을 기준으로 하여, 그 시간에 생성 및 삭제된 파일 리스트 확인 필요
- 특히, self-deletion 기능이 있는 경우에는 악성관련 정보가 파일 시스템이 존재하지 않지만, 메타데이터 파일에는 흔적이 남아 있음.
- 관련된 메타데이터 파일로는 $MFT와 $USNJRNL파일이 있음
  - $MFT 파일: OFFSET 22~23 지점에 있는 INUSE flag 값이 삭제될 경우, 0x00 값을 갖게 됨.
  - $USNJRNL파일: 침해사고 발생 시점 전 후에 삭제된 파일 흔적을 조사해야 하는데, $USNJRNL파일의 기본 사이즈가 작기 때문에, 최근 며칠 이내의 데이터 밖에 존재하지 않음. 그렇기 때문에 unallocation 영역에 있는 $USNJRNL데이터를 찾아서 확인을 해야 함.

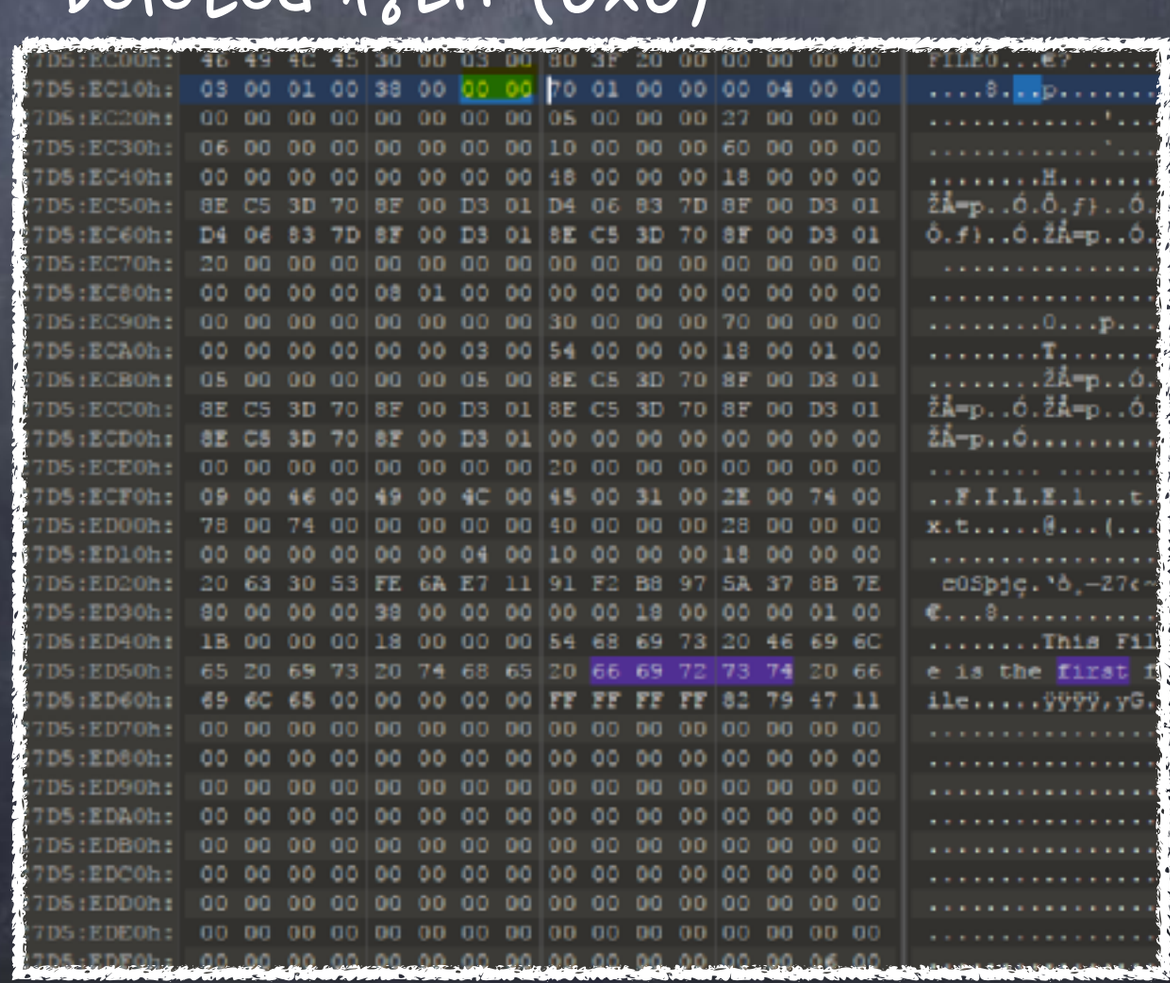
특정 시간에 생성 및 삭제된 파일 정보

$MFT 개별 Entry 별로 OFFSET 22~23 지점의 데이터 값이 0x01이면 Active 상태이며, 0x00이면, deleted 상태를 의미 함.

Active 상태 (0x1)



Deleted 상태 (0x0)



파일 이름 및 사이즈 정보

- 악성코드 유형 중에 자기 자신을 다른 디렉터리에 복제하는 기능이 있는데, 파일명 또는 사이즈 등을 통하여 복제된 파일로 의심할 수 있는 파일을 분류할 수 있음
- 또한, IOC 정보 중에 파일 명과 사이즈 정보가 있다면, 이를 기반으로 하여, $MFT 파일에서 악성코드 의심되는 파일을 분류할 수 있음
- 단, 단순 파일 명과 파일 사이즈 등의 오탐이 발생할 수 있는 indicator 정보이기 때문에, 파일명과 사이즈로 매핑되는 파일정보에 대해서는 추가적인 정보 확인이 필요 함
  - 파일 삭제 되지 않고 파일 시스템에 존재할 경우, 해쉬정보 확인
  - 파일 삭제된 경우, 카빙을 통한 복구 시도
  - 삭제 유무와 상관 없이 해당 파일의 컨텐츠가 $DATA 영역에 존재할 경우, 컨텐츠 확인
- 이는 사고분석 과정이 하나의 퍼즐을 맞추어 가는 과정인 점을 고려했을 때, 다양한 아티 팩트 정보를 수집 및 확인한다는 관점에서 이러한 정보도 의미가 있을 것으로 판단 됨

$DATA 영역에 존재하는 데이터

- $MFT내부에 존재하는 $DATA영역은 파일 사이즈가 작은 경우(예, < 780Byte)는 $DATA에 저장을 함. 이럴 경우, Resident Flag를 설정하여 파일 데이터가 이 영역 저장되어 있다는 것을 알림

- 정상적인 경우 외에, 의도적으로 non-resident로 설정되어 있는 파일 레코드의 $DATA에 데이터를 저장하는 경우가 있음. 이는 data hiding 용도로 사용된 케이스로 의심해야 함.

- non-resident로 설정된 파일 중에 $DATA 영역에 데이터가 있는지 조사하고, 또한 resident로 설정된 파일의 $DATA에서 의미있는 데이터가 있는지 조사 필요