creative commons
COMMONS DEED

저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:

저작자표시. 귀하는 원저작자를 표시하여야 합니다.

비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.

변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 이용허락규약(Legal Code)을 이해하기 쉽게 요약한 것입니다.

Disclaimer

dCollection

이학석사 학위논문

윈도우 색인파일을 활용한 외장저장장치 흔적 분석

2019년 2월

서울대학교 융합과학기술대학원

수리정보과학과 디지털포렌식학 전공

신 재 훈

윈도우 색인파일을 활용한 외장저장장치 흔적 분석

지도교수 천 정 희

이 논문을 이학석사 학위논문으로 제출함
2019년 1월

서울대학교 융합과학기술대학원
수리정보과학과 디지털포렌식학 전공
신 재 훈

신재훈의 석사 학위논문을 인준함
2019년 1월

위 원 장 이 효 원 (인)

부위원장 박 상 준 (인)

위 원 천 정 희 (인)

초 록

디지털포렌식 수사관은 압수수색을 집행하는 과정에서 다양한 상황을 마주하게 된다. 이러한 현장의 상황에 디지털포렌식 수사관이 얼마나 냉철하고 신속하게 대응하는가에 따라 압수수색의 결과는 크게 달라질 것이다.

근래의 압수현장에서는 사용자의 PC에 어떤 외장저장장치가 접속되었으며 그 시점이 언제인지를 판단하는 것이 압수수색의 중요한 키포인트가 되었다. 많은 사람들이 업무와 관련된 자료를 외장하드나 USB 등에 백업하여 보관한다. 사건과 관련한 중요한 자료가 이 외장저장장치에 보관되어 있을 확률이 크기 때문에 수사관은 압수수색 현장에서 사용자가 어떤 외장저장장치를 소유하고 있는지 신속하게 판단하여 그 장치를 확보하여야 한다.

외장저장장치의 흔적을 찾는 방법은 레지스트리 분석이나 링크파일 분석 등 이미 많은 포렌식 분석기법이 알려져 있다. 하지만 윈도우 색인파일인 Windows.edb 파일에 주목한 사람은 많지 않다. 사용자가 특정 볼륨이나 폴더에 색인을 걸었을 경우 이 색인정보가 생성되고, 색인정보를 초기화 시키지 않는 이상 이 정보는 유지되는 특성이 있다. 외장저장장치 역시 색인대상에 포함될 수 있다.

사용자가 색인을 걸었을 경우라는 전제조건이 있기 때문에 이 색인정보는 아주 제한적인 경우에만 활용될 수 있다. 하지만 사용자가 PC를 사용하면서 본인도 모르게 특정 볼륨이나 폴더에 색인을 걸어놓는 경우가 많다. 만약 단 한번이라도 색인을 걸어 놓았

다면 대상 볼륨에 대한 색인이 계속 유지되기 때문에, 이 정보를 토대로 사건의 열쇠가 될 중요한 데이터를 획득하게 될 가능성은 분명히 있다. 수사관의 입장에서는 작은 가능성 하나라도 소홀히 할 수 없다. 이 색인파일에 대한 정보 역시 반드시 확인하여야 할 것이다.

본 논문에서는 윈도우 색인파일인 Windows.edb 파일의 구조에 대하여 간단히 살펴본 후, 외장저장장치의 흔적을 분석하는 기존의 포렌식 아티팩트에 관하여 기술하겠다. 그리고 이 색인파일 내부의 데이터가 어떻게 생성-변경되는지 실험한 후 색인파일의 특성을 도출하겠다. 그리고 이 결과를 토대로 색인파일을 활용하는 방법에 대하여 제안하고자 한다.

주요어 : Windows.edb, Windows Search, 색인정보, 외장저장장치
학 번 : 2017-21469

목 차

1. 서 론 ····· 1

2. 관련 연구 ····· 3

3. 윈도우 색인파일 구조 분석 ····· 5

3.1. 윈도우 색인파일의 정의 ····· 5

3.2. EDB 파일 구조 ····· 8

3.3. EDB Header ····· 10

3.4. Page ····· 12

4. 외장저장장치 사용 흔적 ····· 14

4.1. 외장저장장치 인식 절차 ····· 14

4.2. 외장저장장치 사용 정보 ····· 15

5. 색인파일을 활용한 흔적 조사 ····· 24

5.1. 색인파일에 저장되는 정보 확인 ····· 24

5.2. 윈도우 색인파일 활용 방안 ····· 36

6. 결 론 ····· 47

참고문헌 ····· 49
Abstract ····· 51

표 목 차

[표 1] Windows 버전별 색인파일 저장위치 ····· 8
[표 2] EDB 헤더 포맷 ····· 10
[표 3] EDB 페이지 헤더 포맷 ····· 12
[표 4] 페이지 타입 플래그 ····· 13
[표 5] USB 접속 관련 아티팩트 정리 ····· 22
[표 6] 실험환경 구성 ····· 24
[표 7] 윈도우 버전별 색인 가능 장치 ····· 25
[표 8] 색인파일 분석 프로그램 종류 ····· 25
[표 9] 실험 대상 외부저장장치 ····· 26
[표 10] 로컬PC→외장저장장치 자료 복사 시 색인정보 확인 ·· 27
[표 11] 외장저장장치 자료 수정 시 색인정보 확인 ····· 28
[표 12] 로컬PC 자료 수정 시 색인정보 확인 ····· 36
[표 13] 외장저장장치 로그 비교 ····· 38
[표 14] 로그온/오프 관련 이벤트ID ····· 43
[표 15] 로그온/오프 시간관련 오프셋 정보 ····· 45

그 림 목 차

[그림 1] 탐색기 검색어 입력 화면 ····· 6
[그림 2] 탐색기 색인에 추가 화면 ····· 6
[그림 3] 제어판 색인옵션 화면 ····· 7
[그림 4] EDB 파일 전체 구조 ····· 9
[그림 5] B-Tree 구조 ····· 13
[그림 6] 외장저장장치 인식 과정 ····· 15
[그림 7] Setupapi.dev.log 저장경로 ····· 16
[그림 8] Setupapi.dev.log 내용 확인 ····· 16
[그림 9] USB 장치 연결 시 이벤트 확인 ····· 17
[그림 10] 각 이벤트의 상세정보 확인 ····· 18
[그림 11] USB 장치 제거 시 이벤트 확인 ····· 18
[그림 12] 상세 이벤트 정보 확인 ····· 19
[그림 13] 자료 삭제 후 색인파일 데이터 확인 ····· 30
[그림 14] BCwipe를 이용한 파일 삭제 ····· 30
[그림 15] NTFS 파일시스템에서 와이핑 확인 ····· 31
[그림 16] FAT 파일시스템에서 와이핑 확인 ····· 31
[그림 17] 완전삭제 후 색인파일 확인 ····· 32
[그림 18] 파일생성 후 해당 오프셋 확인 ····· 33
[그림 19] 파일삭제 후 해당 오프셋 확인 ····· 33
[그림 20] 파일복사 후 해당 오프셋 확인 ····· 34
[그림 21] 색인제외 후 색인파일 확인 ····· 35
[그림 22] 외장저장장치 연결 해제-색인옵션 ····· 37
[그림 23] USBOblivion 이용하여 접속기록 삭제 ····· 40
[그림 24] 레지스트리에서 관련 데이터 삭제됨 ····· 40

[그림 25] CFT에서 접속기록 확인불가 ····· 41
[그림 26] Encase에서 접속기록 확인불가 ····· 41
[그림 27] 이벤트뷰어에서 관련 이벤트 확인 ····· 43
[그림 28] 레지스트리에서 계정명 확인 ····· 44
[그림 29] 레지스트리에서 계정 로그온/오프 시간 확인 ····· 44

1. 서 론

디지털포렌식 수사관은 다양한 압수수색 현장을 경험하게 된다. 사건의 유형에 따라서 쟁점이 되는 사안이 다르기 때문에 수사관이 현장에서 중점적으로 분석하는 아티팩트 역시 다를 수밖에 없다. 하지만 모든 현장에서 디지털포렌식 수사관이 필수적으로 확인하여야 할 부분이 있다면 바로 외장저장장치 접속흔적 일 것이다.

근래의 압수수색 현장에서는 사용자의 PC에 어떤 외장저장장치가 접속되었으며 그 시점이 언제인지를 판단하는 것이 중요한 압수수색 키포인트가 되었다. 많은 사람들이 업무와 관련된 자료를 외장하드나 USB 등에 백업하여 보관한다. 사건과 관련한 중요한 자료가 이 외장저장장치에 보관되어 있을 확률은 매우 높다. 실제로 압수수색 현장에서 피압수자가 중요한 자료가 담긴 USB를 은폐하려고 시도하다가 적발되는 일이 많이 발생한다. 수사관은 압수수색 현장에서 사용자가 어떤 외장저장장치를 소유하고 있는지 신속하게 판단하여 그 장치를 확보하여야 한다.

사용자가 PC에 외장저장장치를 접속하게 되면 이런 정보는 Setupapi.dev.log 정보나 이벤트로그 정보, 레지스트리 정보 등으로 PC에 기록된다. 수사관은 포렌식 툴을 이용하여 이런 정보를 분석하고 이를 토대로 사용자의 PC에 접속하였던 외장저장장치의 흔적을 확인할 수 있다. 이런 외장저장장치의 흔적과 관련한 분석기법들은 이미 포렌식 분야에서는 널리 알려져 있다.

윈도우 색인파일인 Windows.edb 파일에서도 외장저장장치의 흔적을 찾을 수 있다. 이 파일은 사용자가 검색을 더욱 빨리 할 수 있도록 색인정보를 저장한 파일이다. 윈도우 비스타 버전부터 기본기능으로 탑재되었다. 이 색인파일 안에는 색인된 파일의 이름과, 시간정보, 경로 정보들이 기록되기 때문에 이 자체로 중요한 로그 정보가 된다. 사용자가 볼

륨이나 폴더에 색인을 걸었을 경우에 이 정보가 생성되고 색인파일을 초기화 시키지 않으면 계속 유지되는 특성이 있다. 색인파일은 용량의 제한이 없기 때문에 사용자가 한번이라도 색인을 걸었을 경우 이 정보에서 중요한 데이터를 찾을 확률이 높다. 하지만 사용자가 색인을 걸지 않았다면 이 파일에서는 아무런 데이터도 건지지 못한다.

작은 정보 하나라도 소홀히 다룰 수 없는 수사관의 입장에서는 이 정보를 한번쯤 살펴보고 지나가야 할 필요성이 분명히 있다. 무엇보다도 색인대상으로 외장저장장치를 선택 할 수 있다는 것 때문에 이 색인정보에서 외장저장장치의 흔적을 찾을 가능성이 있다는 것에 주목하여야 한다.

Windows.edb 파일과 관련된 기존의 연구들은 주로 데이터베이스의 포맷을 분석하고 이 스키마를 분석하는 방법에 대하여 논의되었고, 최근에는 B-Tree 구조로 인하여 삭제된 레코드가 남는다는 특성을 발견하여 삭제레코드를 복구하는 기법에 대한 연구가 진행되었다.

본 논문에서는 색인파일 자체에 대한 분석이 아닌 이 파일 내부에 저장된 색인정보를 활용하여 외장저장장치 접속 흔적에 대한 로그로 활용하는 방안에 대하여 제시하고자 한다.

3장에서는 기존의 논문을 참고하여 색인파일의 구조를 살펴보도록 하겠다. 그리고 4장에서는 외장저장장치가 윈도우 OS에 접속할 때 생기는 이벤트를 확인하고 이런 이벤트가 어떤 PC에 어떤 흔적을 남기는지 살펴보겠다. 5장에서는 다양한 장치에 색인파일을 구성한 뒤 상황에 따라 색인파일에 저장되는 정보가 어떻게 변하는지 확인하는 실험을 진행하고 이 결과를 토대로 색인파일을 분석에 활용하는 방법을 기술하겠다.

2. 관련 연구

Gordon은 “A forensic examination of windows desktop search (version 3),”에서 Windows 레지스트리의 엔트리, 트랜잭션 파일에 대한 설명, 데이터베이스 레코드를 저장하는 데 사용되는 데이터 형식에 대한 개요 등을 정리하고 ESE 및 Windows Search에서 사용되는 아티팩트를 연구하였다.[1)]

Douglas는 “Forensic artefacts present in microsoft windows desktop search,”에서 마이크로소프트 ESE API를 사용하여 Desktop Search 내부의 데이터를 추출할 수 있음을 기술하였다. 이 추출용 소프트웨어는 더 이상 사용되지 않지만, API를 이용하여 윈도우 색인파일에서 정보를 추출하는 방법을 제시하였다는 것에 의의가 있다.[2)]

Metz는 “Windows Search: Analysis of the Windows Search Database” 와 “Extensible Storage Engine (ESE) Database File (EDB) format specification”를 통하여 EDB 파일이 포렌식 관점에서 관련 정보의 중요한 소스가 될 수 있음을 밝히고 ESE 데이터베이스 파일의 전체적인 포맷을 분석하였고, Windows Search에서 사용되는 내부구조와 스키마를 분석하였다.[3)]

---

1) JM. Gordon, “A forensic examination of windows desktop - 45 search (version 3)”, Master’s Thesis, Cranfield University, 2009
2) J. Douglas, “Forensic artefacts present in microsoft windows desktop search,” Master’s Thesis, Cranfield University, 2009
3) J. Metz, "Extensible Storage Engine Database File format specification",(https://github.com/libyal/libesedb/blob/master/documentation/Extensible%20Storage%20Engine%20(ESE)%20Database%20File%20(EDB)%20format.asciidoc)

H. Chivers와 C. Hargreaves는 “Forensic data recovery from the windows search database”를 통하여 파일시스템에서 파일이 삭제되면 색인파일 데이터베이스에서도 해당레코드가 삭제되지만 B-Tree 구조로 인해 삭제된 레코드에 정보가 남아 있을 수 있다는 것을 확인하고 기존의 API방식으로 EDB 파일을 추출하는 경우에는 삭제된 레코드 정보는 추출할 수 없기 때문에 카빙하는 방식으로 이 정보를 추출할 수 있음을 제시하였다.[4)]

김정현, 최종현, 이상진은 “ESE 데이터베이스 내의 삭제된 레코드 복구 기법”을 통하여 기존의 ESE 데이터베이스 뷰어의 문제점을 확인하고, 범용적으로 적용할 수 있는 삭제된 레코드 복구 기법을 제시하였다.[5)]

EDB 파일과 관련된 기존의 논문들은 EDB 파일의 구조를 분석하는 방법에 대한 연구가 주로 이루어졌고, 최근에는 파일의 구조적 특성을 이용하여 삭제된 레코드를 복구하는 기법에 대한 연구가 이루어졌다.

본 논문에서는 수사 관점에서 이 색인파일에 저장된 데이터를 분석하고, 이 색인파일을 활용하여 사용자가 소유한 외장저장장치의 흔적을 찾는 방법을 제시하고자 한다.

---

4) H. Chivers and C. Hargreaves, "Forensic data recovery from the windows search database", Digital Investigation, vol. 7, no. 3-4, p. 114-126, Apr. 2011.
5) 김정현, 최종현, 이상진, “ESE 데이터베이스 내의 삭제된 레코드 복구기법” 정보보호학회논문지, 25(5), 1143-1151. Figure 1. 2015.

3. 윈도우 색인파일 구조 분석

윈도우 검색은 사용자가 색인옵션을 사용하여 탐색기를 이용하여 특정파일을 빠르게 검색할 수 있도록 도와주는 기능이다. 비스타 이후의 윈도우 운영체제에 기본기능으로 탑재되어 있으며 색인된 정보는 모두 윈도우 색인파일인 Windows.edb 파일에 저장된다. 이번 장에서는 Windows.edb 파일이 어떤 구조로 구성되어 있는지 살펴보도록 하겠다.

3.1. 윈도우 색인파일의 정의

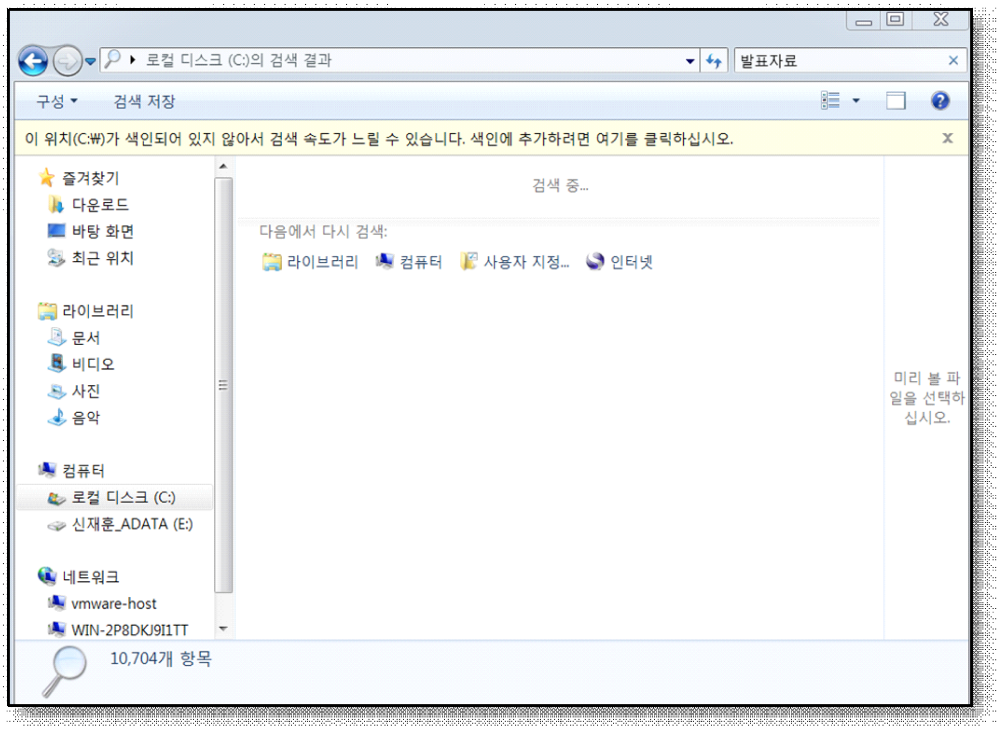
윈도우에서 검색을 빠르게 하기 위하여 색인 정보를 저장하는 파일이다. Microsoft Windows Search(윈도우 검색)는 데이터베이스 인프라와 사용자 인터페이스를 통해 파일, 프로그램, 전자 메일 및 인터넷 기록 레코드를 검색 할 수 있다. 파일은 표준화 된 파일 메타 데이터, 파일 및 날짜, 시간 정보, 파일의 메타 데이터를 포함하는 파일을 생성한다.[6)]

이 색인파일은 'The Extensible Storage Engine (ESE) Database File' 형식으로 구성되어 있다. 이 EDB 파일은 JET Blue 라고도 불리며, 마이크로소프트가 1994년부터 개발한 엔진으로 인터넷 익스플로어, 윈도우 서치, 윈도우 라이브 메일 등에서 폭넓게 사용되고 있다.

윈도우를 사용하면서 사용자는 알게 모르게 PC에 저장된 많은 파일을 검색하게 된다. 빠른 속도로 윈도우의 파일을 검색해 주는 프로그램은 많지만 아마 많은 사용자들이 윈도우의 기본 프로그램인 탐색기를 이용해서 파일을 검색 할 것이다. 탐색기의 우측 상단의 검색 창에 자신이 찾고자 하는 파일명을 입력하였을 때 색인이 걸려 있지 않은 경우 아래의 [그림1]과 같은 화면을 마주하게 된다.

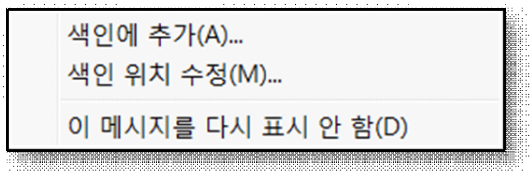
---

6) H. Chivers and C. Hargreaves, "Forensic data recovery from the windows search database," Digital Investigation, vol. 7, no. 3-4, pp. 114-126, Apr. 2011.



[그림 1. 탐색기 검색어 입력 화면]

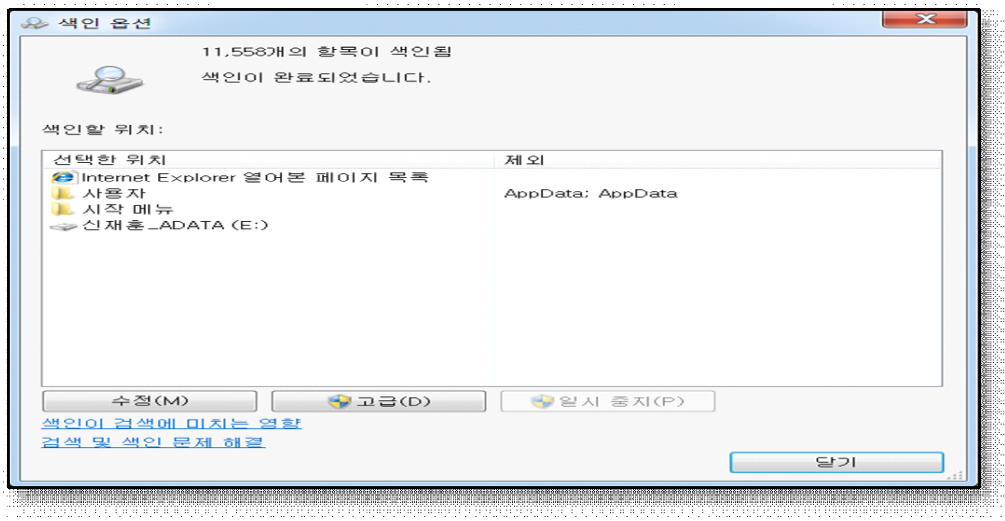
그리고 사용자는 이어서 나오는 [그림2]의 색인에 추가 옵션을 선택할 수 있다.



[그림 2. 탐색기 색인에 추가 화면]

일단 이 색인에 추가 옵션을 선택하면 이후 탐색기의 창을 닫더라도 백그라운드에서 색인 작업을 계속 수행하게 된다. 사용자가 컴퓨터를 끄더라도 색인 작업이 완료되지 않았다면 재부팅 이후 색인 작업은 계속 수행된다. 이와 같은 색인 작업은 윈도우 제어판의 색인 옵션을 통해서도 설정할 수 있다.

제어판의 색인옵션을 선택하면 현재 색인이 되어 있는 위치와 현재 색인 진행 상태를 확인할 수 있다. 외장저장장치의 경우 연결이 해제되어 있다면 이 옵션창의 선택한 위치에 표시되지 않는 것에 주의하여야 한다.



[그림 3. 제어판 색인옵션 화면]

이런 색인정보들은 모두 특정 위치에 Windows.edb 이란 파일로 저장된다. 색인기능은 VISTA에서부터 윈도우에 기본기능으로 탑재되었다. 대부분의 사용자들은 PC를 사용하면서 한번 씩은 검색을 하게 되며, 무심코 색인위치에 추가 버튼을 클릭하게 될 수 있다. 그래서 본인도 모르게 특정 볼륨이나 폴더가 색인되어 있는 경우가 많다. 색인파일에는 파일의 경로와 파일명, 확장자명, 파일의 MAC타임, 색인에 수집된 시간 등의 정보가 기록된다.

색인이 저장된 폴더에는 Windows.edb 파일과 temp 파일이 저장되어 있다. 이 temp 파일들은 PC를 종료하면서 Windows.edb 파일에 업데이트 된다. 색인 파일은 시스템 파일이며, 사용자가 임의로 접근하거나 수정, 삭제 할 수 없다. 색인이 저장된 된 기본 폴더는 [표1]과 같이 윈도우 버전별로 상이하며, 이 저장위치는 사용자가 색인옵션에서 수정할 수 있다.

윈도우 버전	파일 위치
XP	C:\Documents and Sengs\Applicaon Data\AllUsers\Microso\Search\Data\Applicaons\Windows\
Vista/7/8/10	C:\ProgramData\Microso\Search\Data\Applicaons\Windows\

[표 1. Windows 버전별 색인파일 저장위치]

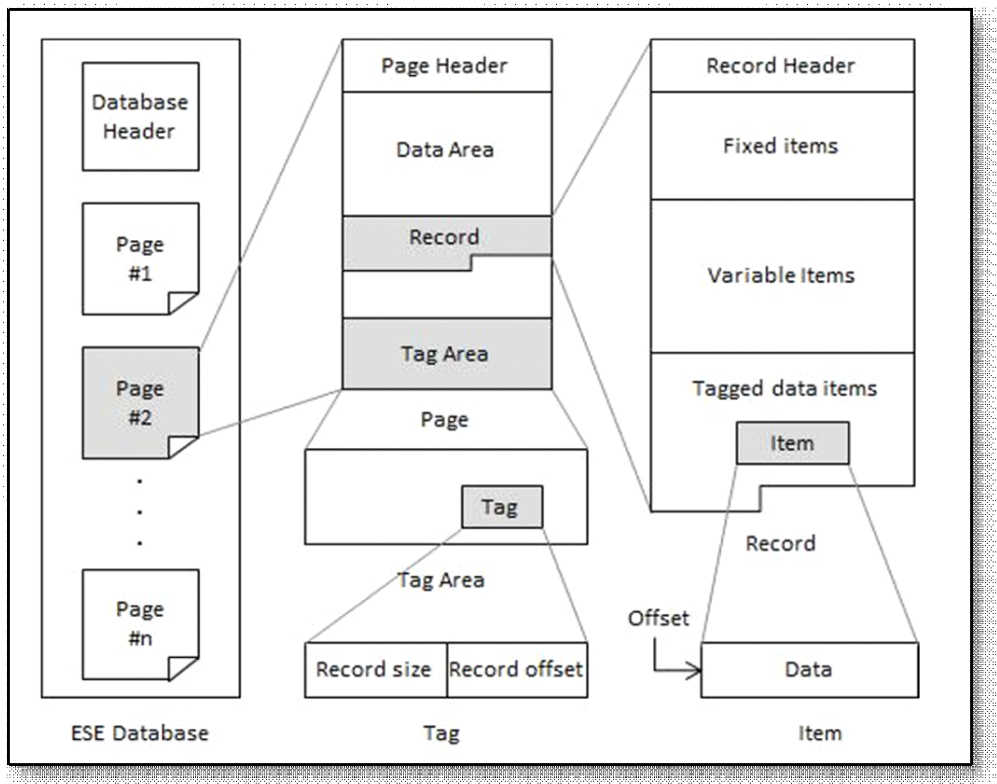
3.2. EDB 파일 구조

Windows.edb 파일은 Extensible Storage Engine(ESE) database 구조로 이루어져 있다. 이 데이터베이스 엔진은 Microsoft사가 개발한 Indexed Sequential Aces Method (ISAM) 의 데이터 저장 방식이다.

ESE 데이터베이스 구조에 대한 연구는 대표적으로 Metz(2010)의 “Windows Search: Analysis of the Windows Search Database” 와 “Extensible Storage Engine (ESE) Database File (EDB) format specification” 그리고 김정현의 “Digital Forensic Investigation of ESE Database” 등이 있다. 본 장에서는 기존에 연구된 논문을 참고하여 ESE 데이터베이스의 구조를 간략히 살펴보도록 하겠다.

Windows.edb 파일이 데이터를 저장하는 단위를 페이지라고 한다. 페이지 사이즈는 윈도우 버전별로 다르게 사용된다. 윈도우7에서 기본으로 사용하는 페이지 사이즈는 16KB 혹은 32KB이다. 데이터베이스 레코드는 페이지 내부에 저장된다. 긴 사이즈의 레코드를 제외하고는 모든 레코드는 단일 페이지 내에 저장된다.[7)]

7) H. Chivers and C. Hargreaves, “Forensic data recovery from the windows search database,” Digital Investigation, vol. 7, no. 3-4, pp. 114-126, Apr. 2011.



[그림 4. EDB 파일 전체 구조[8)]]

위의 [그림4]은 EDB 파일 내부의 전체 구조를 도식화 한 것이다. 색인파일은 윈도우 상에서는 1개의 시스템파일에 불과하지만, 그 내부에는 일정크기의 페이지 단위로 데이터가 구성되며 페이지 안에는 레코드 단위로 정보가 구분되어 기록되는 것을 확인 할 수 있다.

ESE 데이터베이스에는 여러 개의 테이블이 있으며 테이블 정보는 MSysObject 라는 카탈로그 테이블에 의해 관리된다. 이 카탈로그 테이블은 페이지 4번에 저장되어 있으며, ObjidTable, Type, ID 등의 정보로 구성된다.

8) 김정현, 최종현, 이상진, “ESE 데이터베이스 내의 삭제된 레코드 복구기법” 정보보호학회논문지, 25(5), 1143-1151. Figure 1. 2015.

특정 테이블에는 대용량 데이터를 저장하기 위해 LV라는 서브 테이블이 포함된다. 표에는 FDP (Father Data Page) 라고 하는 고유 한 식별 번호가 있는데 이것은 테이블의 모든 페이지 헤더에 지정된다. LV라는 서브 테이블은 또한 고유한 FDP 번호를 사용하며 상위 테이블과 동일한 방식으로 관리된다.[9)]

3.3. EDB Header

ESE database Header는 데이터베이스의 첫 번째 페이지에 저장된다. 668 바이트 사이즈의 정보들로 구성되어 있으며 헤더의 정보를 도식화하면 아래의 [표2]와 같다.

<table>
<tr><th></th><th>0x00</th><th>0x01</th><th>0x02</th><th>0x03</th><th>0x04</th><th>0x05</th><th>0x06</th><th>0x07</th><th>0x08</th><th>0x09</th><th>0x10</th><th>0x11</th><th>0x12</th><th>0x13</th><th>0x14</th><th>0x15</th></tr>
<tr><td>0x0000</td><td colspan="4">Checksum</td><td colspan="4">The Signature</td><td colspan="4">File format version</td><td colspan="4">File type</td></tr>
<tr><td>0x0016</td><td colspan="8">Database time</td><td colspan="8"></td></tr>
<tr><td>0x0032</td><td colspan="8" rowspan="2">Database signature</td><td colspan="8"></td></tr>
<tr><td>0x0048</td><td colspan="4">Database state</td><td colspan="4">Consistent position</td></tr>
<tr><td>0x0064</td><td colspan="8">Consistent date and time</td><td colspan="8">Attach date and time</td></tr>
<tr><td>0x0080</td><td colspan="8">Attach position</td><td colspan="8">Detach date and time</td></tr>
<tr><td>0x0096</td><td colspan="8">Detach position</td><td colspan="8"></td></tr>
<tr><td>0x0112</td><td colspan="8" rowspan="2">Log signature</td><td colspan="8"></td></tr>
<tr><td>0x0128</td><td colspan="4">Unknown</td><td colspan="4"></td></tr>
<tr><td>0x0144</td><td colspan="16">Previous full backup</td></tr>
<tr><td>0x0160</td><td colspan="16">Previous incremental backup</td></tr>
</table>

9) 김정현, 최종현, 이상진, “ESE 데이터베이스 내의 삭제된 레코드 복구기법” 정보보호학회논문지, 25(5), 1143-1151. Figure 1. 2015.

0x0176				
0x0192	Current full backup			
0x0208	Shdowing disabled	Last object identifier	Major version	Minor version
0x0224	Build number	Service pack number	File format revision	Page size
0x0240	Repair count	Repair date and time		
0x0256	Unknown			
0x0272			Scrub database tume	
0x0288	Scrub data and time		Required log	
0x0304	Upgrade Exchange 5.5 format	Upgrade Free pages	Upgrade Space Map pages	
0x0320	Current Shadow copy backups			
0x0336		Creation file format version	Creation file format revision	Unknown
0x0352	Unknown			Old repair count
0x0368	ECC fix success count	Last ECC fix success date and time		Old ECC fix success count
0x0384	ECC fix error count	Last ECC fix error date and time		Old ECC fix error count
0x0400	Bad checksum error count	Last bad checksum error date and time		Old bad checksum error count
0x0416	committed log	Previous copy backup		
0x0432				
0x0448	previous differential backup			
0x0464		unknown		
0x0480				
0x0496				NLS major version
0x0512	NLS minor version			
0x0528	Unknown			
0x0544				

[표 2. EDB 헤더 포맷]

ESE database Header의 데이터는 리틀엔디안 방식으로 저장되어 있다. 첫 4바이트는 Checksum 정보가 기록되어 있으며, 뒤의 4바이트는 ESE database Header의 고유한 시그니처(\xEF\xCD\xAB\x89)가 기록되어 있다. 데이터베이스의 상태 (Dirty Shutdown 혹은 Clean Shutdown) 정보와 데이터베이스 관련 시간정보, 페이지 사이즈 등 데이터베이스에 대한 전반적인 정보가 기록되어 있다. 데이터베이스 헤더에는 6바이트의 시간정보들이 기록되어 있는데 이 시간정보는 앞의 각 1바이트부터 Second, Minute, Hour, Day, Month, Year(since1900) 순으로 해석할 수 있다.

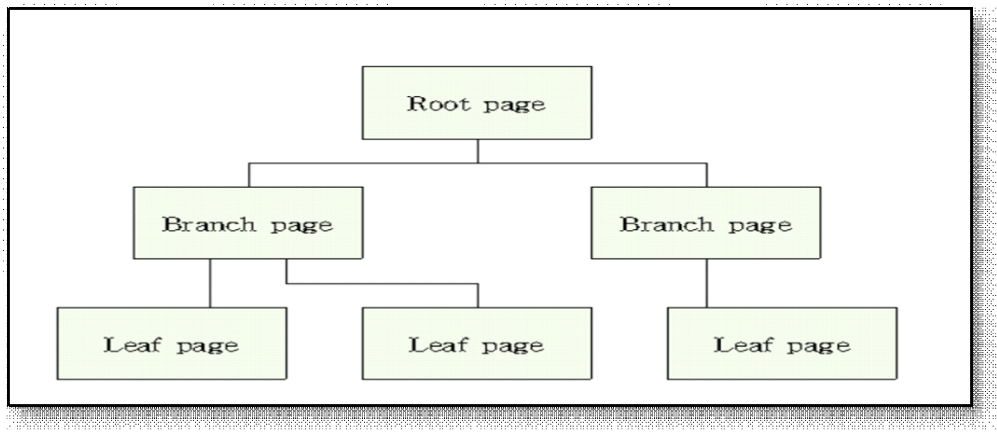
3.4. Page

EDB 데이터베이스에서 데이터를 저장하고 관리하는 고정된 단위를 페이지라고 부른다. 페이지의 사이즈는 데이터베이스 헤더에서 정의된다. 페이지는 페이지 헤더와 페이지 태그 그리고 페이지 데이터 영역으로 나뉜다. 각 페이지에는 기본적으로 80바이트의 페이지 헤더정보가 기록되어 있으며 이를 도식화 하면 아래의 [표3]과 같다. 페이지 헤더에는 페이지에 대한 전반적인 정보가 담겨 있다.

<table>
<tr><th></th><th>0x00</th><th>0x01</th><th>0x02</th><th>0x03</th><th>0x04</th><th>0x05</th><th>0x06</th><th>0x07</th><th>0x08</th><th>0x09</th><th>0x10</th><th>0x11</th><th>0x12</th><th>0x13</th><th>0x14</th><th>0x15</th></tr>
<tr><td>0x0000</td><td colspan="8">XOR checksum</td><td colspan="8">Database last modification time</td></tr>
<tr><td>0x0016</td><td colspan="4">Previous page number</td><td colspan="4">Next page number</td><td colspan="4">FDP object identifier</td><td colspan="2">Available data size</td><td colspan="2">Available uncommitted data size</td></tr>
<tr><td>0x0032</td><td colspan="2">Available data offset</td><td colspan="2">Available page tag</td><td colspan="4">Page flags</td><td colspan="8">Extended checksum 1</td></tr>
<tr><td>0x0048</td><td colspan="8">Extended checksum 2</td><td colspan="8">Extended checksum 3</td></tr>
<tr><td>0x0064</td><td colspan="8">Page number</td><td colspan="8">Unknown(Empty values)</td></tr>
</table>

[표 3. EDB 페이지 헤더 포맷]

모든 페이지들은 B-Tree 구조로 관리된다. 각 페이지들은 Root page, Branch page, Leaf Page로 구분된다. B는 Balance를 의미하며, 실제 데이터는 Leaf page에 존재한다. 가장 상위에 위치하는 Root Page 부터 실제 데이터가 존재하는 Leaf page까지 각 노드들의 값을 동일하게 만들어서 빠르고 효과적인 데이터 접근이 가능하다. 아래의 [그림5] 은 B-Tree의 구조를 표현한 것이다.



[그림 5. B-Tree 구조]

페이지 헤더에는 페이지가 어떤 타입인지를 나타내는 페이지 플래그 정보가 담겨 있다. 이 플래그 정보를 도식화 하면 다음과 같다.

Value	Description
0x00000001	This page is a root page
0x00000002	This page is a Leaf page
0x00000004	This page is a Parent page
0x00000008	This page is a space tree page
0x00000020	This page is a index page
0x00000040	This page is a Long value page
0x00000400	Unknown
0x00000800	Unknown
0x00002000	New record format

[표 4. 페이지 타입 플래그]

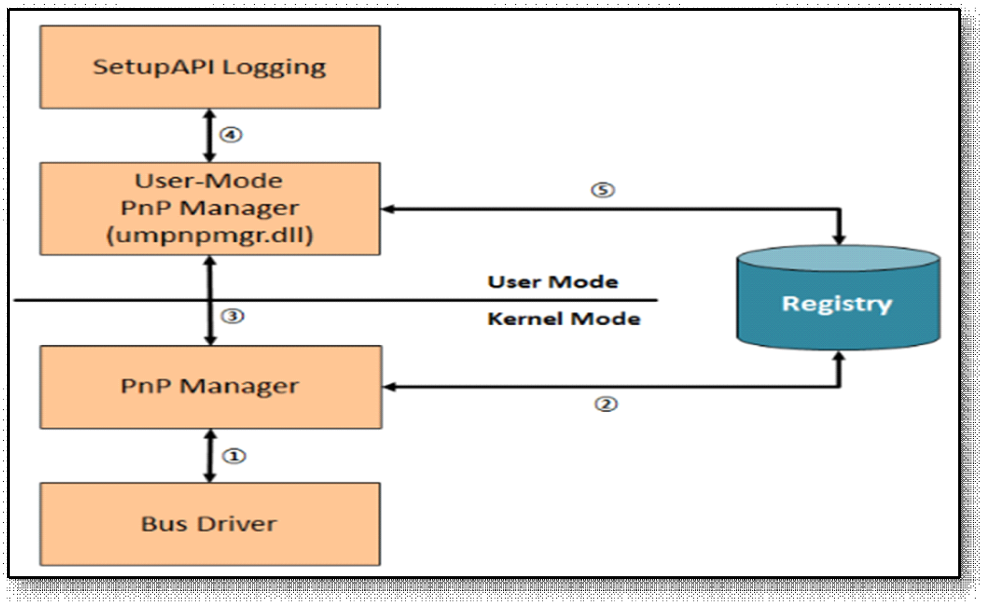
각 페이지의 레코드 저장방법은 페이지의 타입에 따라 다르다. 이는 data 페이지, branch 페이지, long value 페이지를 분석하여 식별할 수 있다. 페이지 태그는 페이지에 저장된 각 레코드의 오프셋위치와 크기를 기록한다. 페이지 태그는 페이지의 끝부분에 저장이 된다. 끝에서부터 앞쪽으로 순차적으로 저장되는 특징이 있다. 페이지 내에 존재하는 페이지 태그의 개수는 페이지 헤더에 저장되어 있다.

4. 외장저장장치 사용 흔적

이번 장에서는 USB 디바이스의 접속 흔적을 분석할 수 있는 기본적인 아티팩트를 살펴보겠다.

4.1. 외장저장장치 인식 절차

윈도우 시스템에 USB 디바이스가 최초로 연결되면 버스드라이버가 PnP 관리자에 장치의 고유한 식별 정보를 보낸다. 커널모드의 PnP 관리자는 식별정보를 기반으로 레지스트리를 확인한 후 드라이버가 이미 설치되어 있다면 이를 로드하고, 드라이브가 없을 경우 장치 펌웨어에게 드라이버를 요청한다. 이후 전달받은 드라이버를 사용자 모드 PnP 관리자에게 전달한다. 사용자 모드 PnP 관리자는 전달받은 드라이브를 설치하고, 이후 설치가 완료되면 USB 장치를 시스템에 마운트한다. 아래의 [그림6]은 이와 같은 정보를 도식화 한 것이다. 결과적으로 이와같은 외장저장장치의 인식과정을 통하여 Setupapi.dev.log 와 레지스트리, 이벤트로그에 각각 접속흔적을 남기게 된다.



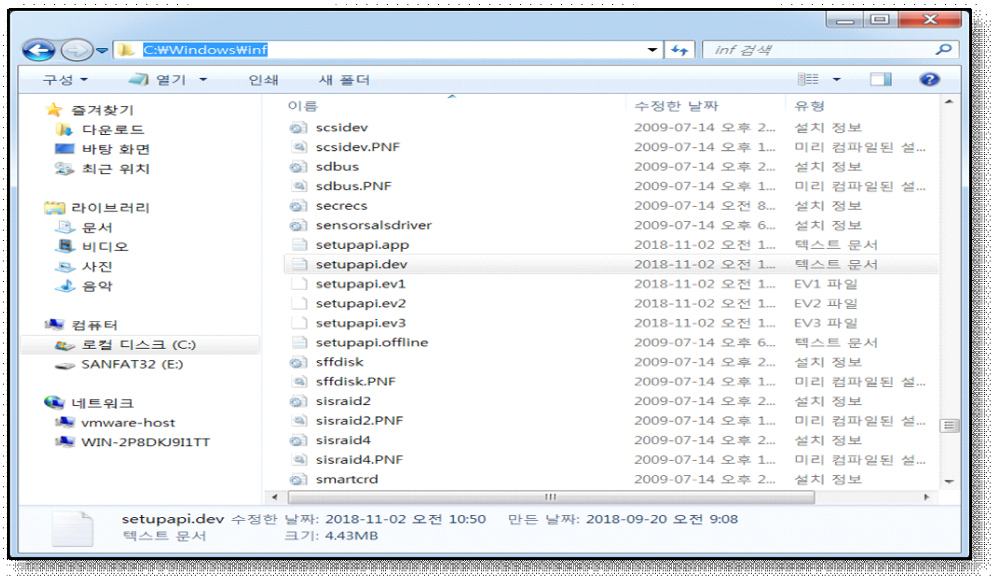
[그림 6. 외장저장장치 인식 과정][10]

4.2. 외장저장장치 사용 정보

(1) Setupapi.dev.log 정보

서비스 팩이나 장치 드라이버가 설치될 때 남기는 텍스트 로그 정보이다. USB 디바이스 최초 연결 시에만 장치 드라이버 설치과정이 저장된다. <c:\windows\inf\Setupapi.dev.log> 의 경로에 저장되며 로그파일에서 장치의 제품명이나 시리얼번호를 검색하여 해당 장치의 연결 사실과 최초 연결 시각을 확인할 수 있다.

10) http://forensic-proof.com/archives/3632



[그림 7. Setupapi.dev.log 저장경로]

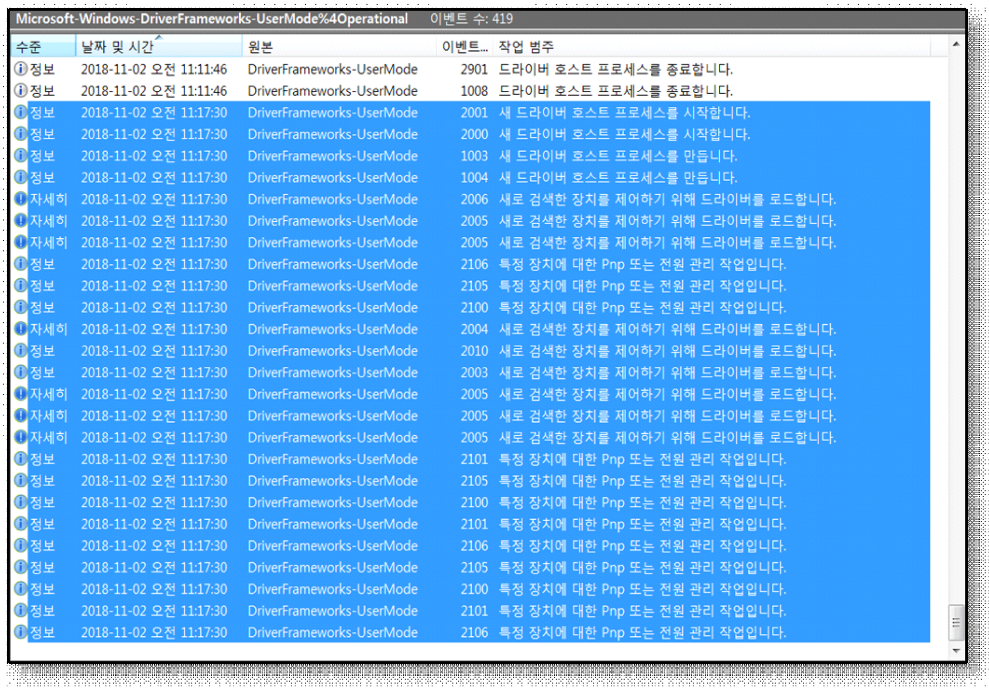
```
[Device Install (Hardware initiated) - STORAGE₩Volume₩_??_USBSTOR#Disk&Ven_SanDisk&Prod_Ultra&Rev_1.00
Section start 2018/11/02 10:50:38.576
ump:  Creating Install Process: DrvInst.exe 10:50:38.608
ndv:  Retrieving device info...
ndv:  Setting device parameters...
ndv:  Searching Driver Store and Device Path...
dvi:  {Build Driver List} 10:50:38.608
dvi:       Searching for hardware ID(s):
dvi:            storage₩volume
cpy:       Policy is set to make all digital signatures equal.
dvi:       Enumerating INFs from path list 'C:₩Windows₩inf'
inf:       Opened PNF: 'C:₩Windows₩System32₩DriverStore₩FileRepository₩volume.inf_amd64_neutral_1b1a512
dvi:       Created Driver Node:
dvi:            HardwareID    - STORAGE₩Volume
dvi:            InfName       - C:₩Windows₩System32₩DriverStore₩FileRepository₩volume.inf_amd64_neutral_
dvi:            DevDesc       - 일반 볼륨
dvi:            DrvDesc       - 일반 볼륨
dvi:            Provider      - Microsoft
dvi:            Mfg           - Microsoft
dvi:            ModelsSec     - MSFT.NTamd64
dvi:            InstallSec    - volume_install
dvi:            ActualSec     - volume_install.NTAMD64
dvi:            Rank          - 0x00ff0000
dvi:            Signer        - Microsoft Windows
dvi:            Signer Score  - INBOX
dvi:            DrvDate       - 06/21/2006
dvi:            Version       - 6.1.7600.16385
```

[그림 8. Setupapi.dev.log 내용 확인]

[그림8]의 Device Install 부분을 보면 제조사(SanDisk)와 제품명(Ultra)을 확인할 수 있고, Section start 부분을 보면 USB가 2018. 11. 2. 10:50경 처음 연결된 것을 확인할 수 있다.

(2) 이벤트 로그 정보

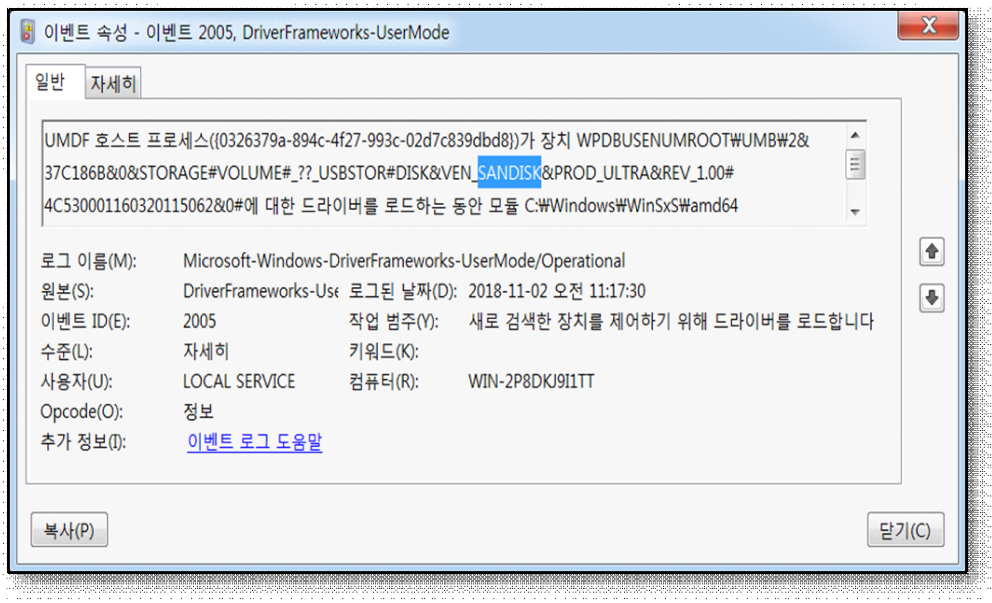
USB 디바이스 등의 장치를 연결하면 드라이브가 로드되면서 이벤트 로그가 기록된다. 이 정보는 <C:\windows\config\System32\winevt\ Logs\Microsoft-Windows-DriverFrameworks-UserMode%4Operational.evtx> 의 경로에 저장 된다. 1003, 2003, 2010, 2004 등 다양한 이벤트 ID를 가지며 모두 장치의 이름이나 ID 정보를 가지고 있다. 각 이벤트에는 제조사, 제품명, 시리얼 정보 등이 기록되는 장치식별자가 존재한다. 다음의 [그림9]는 장치가 연결되었을 때 25개의 접속이벤트를 확인한 것이다.



Microsoft-Windows-DriverFrameworks-UserMode%4Operational 이벤트 수: 419

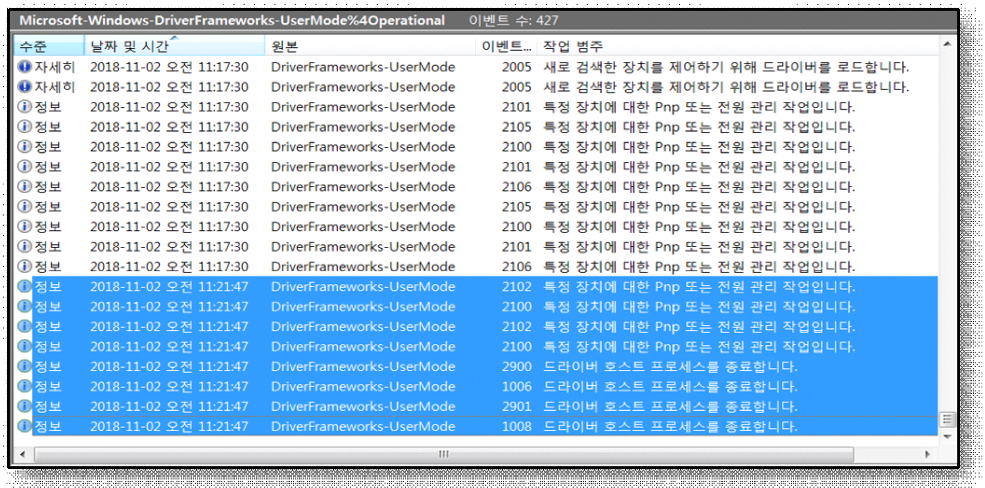
수준	날짜 및 시간	원본	이벤트...	작업 범주
정보	2018-11-02 오전 11:11:46	DriverFrameworks-UserMode	2901	드라이버 호스트 프로세스를 종료합니다.
정보	2018-11-02 오전 11:11:46	DriverFrameworks-UserMode	1008	드라이버 호스트 프로세스를 종료합니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2001	새 드라이버 호스트 프로세스를 시작합니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2000	새 드라이버 호스트 프로세스를 시작합니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	1003	새 드라이버 호스트 프로세스를 만듭니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	1004	새 드라이버 호스트 프로세스를 만듭니다.
자세히	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2006	새로 검색한 장치를 제어하기 위해 드라이버를 로드합니다.
자세히	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2005	새로 검색한 장치를 제어하기 위해 드라이버를 로드합니다.
자세히	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2005	새로 검색한 장치를 제어하기 위해 드라이버를 로드합니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2106	특정 장치에 대한 Pnp 또는 전원 관리 작업입니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2105	특정 장치에 대한 Pnp 또는 전원 관리 작업입니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2100	특정 장치에 대한 Pnp 또는 전원 관리 작업입니다.
자세히	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2004	새로 검색한 장치를 제어하기 위해 드라이버를 로드합니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2010	새로 검색한 장치를 제어하기 위해 드라이버를 로드합니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2003	새로 검색한 장치를 제어하기 위해 드라이버를 로드합니다.
자세히	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2005	새로 검색한 장치를 제어하기 위해 드라이버를 로드합니다.
자세히	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2005	새로 검색한 장치를 제어하기 위해 드라이버를 로드합니다.
자세히	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2005	새로 검색한 장치를 제어하기 위해 드라이버를 로드합니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2101	특정 장치에 대한 Pnp 또는 전원 관리 작업입니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2105	특정 장치에 대한 Pnp 또는 전원 관리 작업입니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2100	특정 장치에 대한 Pnp 또는 전원 관리 작업입니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2101	특정 장치에 대한 Pnp 또는 전원 관리 작업입니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2106	특정 장치에 대한 Pnp 또는 전원 관리 작업입니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2105	특정 장치에 대한 Pnp 또는 전원 관리 작업입니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2100	특정 장치에 대한 Pnp 또는 전원 관리 작업입니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2101	특정 장치에 대한 Pnp 또는 전원 관리 작업입니다.
정보	2018-11-02 오전 11:17:30	DriverFrameworks-UserMode	2106	특정 장치에 대한 Pnp 또는 전원 관리 작업입니다.

[그림 9. USB 장치 연결 시 이벤트 확인]

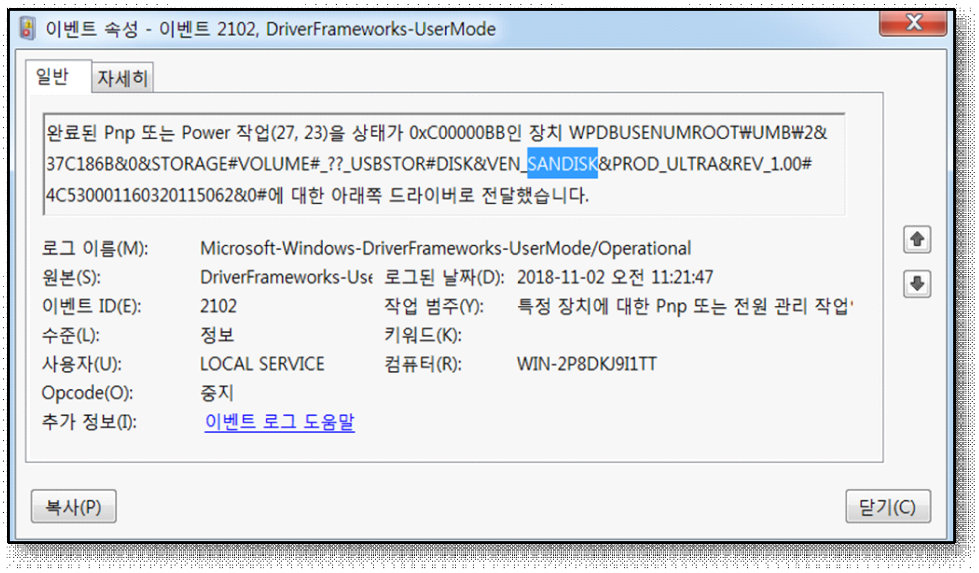


[그림 10. 각 이벤트의 상세정보 확인]

장치가 시스템에서 제거되면 장치의 드라이버 호스트 프로세스가 종료되면서 이벤트 ID 2100, 2102, 1006 등의 정보를 가지는 이벤트가 기록된다.



[그림 11. USB 장치 제거 시 이벤트 확인]



[그림 12. 상세 이벤트 정보 확인]

이와 같이 이벤트로그를 분석하면 장치의 연결 및 해제 흔적을 발견할 수 있다. 반드시 USB 디바이스가 접속할 경우에만 로그가 기록되기 때문에 비교적 신뢰할 수 있는 정보이다. 단 이벤트 로그는 모든 드라이버에 대한 이벤트이기 때문에 USBTOR의 장치식별자만 구분해서 확인할 필요가 있다. 만약 혐의자에 대하여 특정된 USB 정보가 있다면 이 장치가 언제 접속되고 언제 해제되었는지도 추적할 수 있다.

(3) 레지스트리 정보

레지스트리는 마이크로소프트 윈도우 운영체제에서 운영체제와 응용프로그램 운영에 필요한 정보를 저장하기 위해 고안된 계층형 데이터베이스를 말한다.[11] 윈도우 레지스트리는 로그인, 서비스 실행, 응용프로그램, 사용자 행위 등 모든 활동에 관여한다. USB 디바이스의 접속 흔적 역시 레지스트리에 기록된다.

11) https://docs.microsoft.com/ko-kr/windows/desktop/SysInfo/registry

1) Device class ID

접속되었던 USB 디바이스 장치의 형식, 제조사명, 상품명, 버전의 정보를 <Disk&Ven_[제조사]&Prod_[제품명]&Rev_[버전]> 와 같은 형식으로 기록한다.

2) Unique Instance ID

Device class ID의 하위키이다. 장치를 고유하게 식별할 수 있도록 시스템에서 제공한 문자열로서 USB 디바이스에 고유한 시리얼번호가 있다면 이 시리얼 번호가 저장된다. 만약 시리얼번호가 존재하지 않는 USB가 접속한 경우에는 PnP 관리자가 생성한 값이 기록된다.

<고유 시리얼번호 존재 O : (시리얼번호)&#>
<고유 시리얼번호 존재 X : #&(랜덤 번호)&#>

3) 볼륨 레이블

볼륨 레이블은 파일시스템의 특정 볼륨에 주어지는 이름을 말한다. 만약 사용자가 볼륨 레이블을 지정했을 경우에는 이 정보도 레지스트리에 기록된다.

4) 볼륨 시리얼 번호

볼륨 시리얼 번호는 윈도우 운영체제에서 저장장치를 포맷할 때 생성되는 고유한 16진수의 값이다. 볼륨은 하나의 파일시스템을 가지고 있고, 윈도우 파일시스템은 모두 고유한 볼륨 시리얼 번호를 가지고 있다. 대부분의 파일시스템은 VBR(Volume Boot Record)의 부트섹터에 볼륨 시리얼 번호를 기록한다. USB 디바이스를 연결할 때 이 정보를 레지스트리에 남기게 된다.

5) 볼륨 GUID

전역고유식별자 GUID(Globally Unique Identification Number)는 응

용 소프트웨어에서 사용되는 유사 난수를 말한다. GUID는 생성될 때 항상 유일한 값이 만들어진다는 보장은 없지만, 사용할 수 있는 모든 값의 수가 $2^{128} = 3.4028 \times 10^{38}$ 개로 매우 크기 때문에, 같은 숫자를 두 번 생성할 가능성은 매우 낮다. 윈도우는 시스템에 어떤 장치가 연결되면 레지스트리에 해당 장치의 볼륨 GUID값을 생성한다. 하위 키값은 "\??\Volume" 라는 문구 뒤에 GUID 정보가 기록된다.

6) 스마트폰 접속 기록

스마트 폰을 충전하거나 내부 저장소를 저장장치로 이용하기 위하여 PC에 연결하였을 경우에도 그 정보가 레지스트리에 기록된다.

7) 시간정보

레지스트리에는 시간정보를 알 수 있는 키도 존재한다. 레지스트리 키 중에 "HKLM\SYSTEM\ControlSet00#\Enum\USB 또는 USBSTOR" 의 LastWrite 시간의 의미는 불분명하다. Vista/7 이후 보안정책에 의해 PnP 관리자가 수시로 접근하여 시간이 변경될 수 있기 때문에 신뢰적이지 못하다.[12)]

HKLM\SYSTEM\ControlSet00#\Control\DeviceClasses\{53F56307-B6BF-11D0-94F2-00A0C91EFB8B}의 하위키 중 "USBSTOR"를 포함하는 서브키의 LastWrite 시간은 해당 시스템이 마지막으로 부팅한 후 USB 장치가 시스템에 최초로 연결된 시간을 의미한다.

HKLM\SYSTEM\ControlSet00#\Control\DeviceClasses\{53F5630D-B6BF-11D0-94F2-00A0C91EFB8B}의 하위키의 LastWrite 시간은 해당 시스템이 마지막으로 부팅한 후 USB장치 내 볼륨이 시스템에 최초로 연결된 시간을 나타낸다.

아래 [표5]는 외장저장장치와 관련된 아티팩트를 정리한 표이다.

---

12) http://forensic-proof.com/archives/3632

아티팩트	레지스트리 경로
Vendor & ProductName, Version	HKLM\SYSTEM\ControlSet00#\Enum\USBSTOR\Disk&Ven_{Vendor Name}&Prod_{Product Name}&Rev_{Version}
Vendor ID, Product ID	HKLM\SYSTEM\ControlSet00#\Enum\USB\VID_{Vendor ID} & PID_{Product ID}
Serial Number	HKLM\SYSTEM\ControlSet00#\Enum\USB\{Vendor ID &Product ID}\{Serial Number} HKLM\SYSTEM\ControlSet00#\Enum\USBSTOR\{Device Class ID}\{Serial Number}&#
Volume Serial Number	HKLM\SOFTWARE\Microsoft\WindowsNT\CurrentVersion\EMDMgmt_??_USBSTOR#{Device Class ID}#{Unique Instance ID}#{GUID}{Volume Label}_{Volume Serial Number}
Volume Label	HKLM\SOFTWARE\Microsoft\WindowsNT\CurrentVersion\EMDMgmt_??_USBSTOR#{Device Class ID}#{Unique Instance ID}#{GUID}{Volume Label}_{Volume Serial Number} HKLM\SOFTWARE\Microsoft\Windows Portable Devices\Devices\{Device Entry}\FriendlyName (value) HKLM\System\ControlSet00#\Enum\WpdBusEnumRoot\UMB\{Device Entry}\FriendlyName (value)
Drive Letter	HKLM\System\MountedDevices (search for serial number) HKLM\SOFTWARE\Microsoft\Windows Portable Devices\Devices\{Device Entry}\FriendlyName (value) HKLM\System\ControlSet00#\Enum\WpdBusEnumRoot\UMB\{Device Entry}\FriendlyName (value)
Volume GUID	HKLM\SYSTEM\MountedDevices\\??\Volume{Volume GUID} (search for serial number)

User Name	HKU\{USER}\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2\{Volume GUID}
First Connection Time (Last Written Time in registry key)	%SystemRoot%\inf\Setupapi.dev.log HKLM\SYSTEM\ControlSet00#\Control\DeviceClasses\{10497B1B-BA51-44E5-8318-A65C837B6661}\{Sub Keys} HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\EMDMgmt\{Device Entry} HKLM\SOFTWARE\Microsoft\Windows Portable Devices\Devices\{Device Entry}
First Connection Time After Booting (Last Written Time in registry key)	HKLM\SYSTEM\ControlSet00#\Control\DeviceClasses\{53F56307-B6BF-11D0-94F2-00A0C91EFB8B}\{Sub Keys} HKLM\SYSTEM\ControlSet00#\Control\DeviceClasses\{53F5630D-B6BF-11D0-94F2-00A0C91EFB8B}\{Sub Keys} HKLM\SYSTEM\ControlSet00#\Control\DeviceClasses\{6AC27878-A6FA-4155-BA85-F98F491D4F33}\{Sub Keys} HKLM\SYSTEM\ControlSet00#\Control\DeviceClasses\{A5DCBF10-6530-11D2-901F-00C04FB951ED}\{Sub Keys} HKLM\SYSTEM\ControlSet00#\Enum\WpdBusEnumRoot\UMB\{Device Entry}
Last Connection Time (Last Written Time in registry key)	HKU\{USER}\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2\{Volume GUID} HKLM\SYSTEM\ControlSet00#\Control\DeviceClasses\{53F5630D-B6BF-11D0-94F2-00A0C91EFB8B}{Sub Keys}

[표 5. USB 접속 관련 아티팩트 정리(win7)][13]

13) http://forensic-proof.com/archives/3632

5. 색인파일을 활용한 흔적 조사

이번 장에서는 색인파일에 색인정보를 구성한 뒤 다양한 조건에서 이 색인정보가 어떻게 변경되는지 실험해 보겠다. 그리고 실험을 토대로 색인정보의 특성을 도출하고, 이 색인파일을 활용하여 외장저장장치의 흔적을 찾는 방법을 제시하도록 하겠다.

5.1. 색인파일에 저장되는 정보 확인

다음 [표6]과 같은 VMware 환경을 구성하고 각 실험단계마다 CFT(ver.3.6.2)[14]를 이용하여 색인파일을 추출하였다. 추출된 색인파일은 ESE DatabaseView (ver1.62)[15] 프로그램을 이용하여 내부 데이터를 확인하였다.

윈도우 버전	Windows10 Home
메모리	2GB
용량	60GB

[표 6 실험환경 구성]

아래의 [표7]과 같이 윈도우 버전에 따라서 색인정보를 구성할 수 있는 외부저장장치가 다른 것을 확인하였다. 스마트폰을 PC에 연결하여 이동식디스크로 인식시키는 경우에는 모든 윈도우 버전에서 색인정보를 구성하는 것이 불가능하였다. 실험은 윈도우10 버전에서 진행하였다.

---

14) Computer Forensics Tool. 국가보안기술연구소에서 개발한 현장압수용 디지털포렌식 프로그램. 대검찰청을 비롯하여 국내 다수의 수사기관에서 사용하고 있다.

15) NirSoft에서 개발한 Extensible Storage Engine (ESE) database 뷰어 프로그램 이다. 프리웨어. https://www.nirsoft.net/utils/ese_database_view.html

윈도우 버전	색인 가능한 외부저장장치
윈도우 7	외장하드
윈도우 8, 10	microSD카드, USB, 외장하드

[표 7. 윈도우 버전별 색인 가능 장치]

윈도우 7, 8, 10 버전에서 확인한바 윈도우에서 디폴트로 설정된 색인 대상 위치는 사용자폴더(user)와 시작메뉴 폴더이다. 이외에는 사용자가 PC를 사용하면서 임의로 지정한 위치가 추가된다. 색인파일을 분석하는 프로그램의 종류는 아래의 [표8]와 같다. 대부분의 프로그램은 삭제레코드를 분석하지 못하거나 색인파일의 크기가 클 경우 메모리 부족으로 프로그램이 강제 종료되는 등의 단점이 있다. 최근에는 이런 문제점을 개선한 WinSearchDBAnalyzer 라는 프로그램이 발표되었다.

분석 프로그램	특징
ESEDatabaseView (NirSoft)	Dirty상태 분석 O, 삭제레코드 분석 X 메모리 부족 강제종료, 프리웨어
EseDbViewer (Woanware)	Dirty상태 분석 X, 삭제레코드 분석 X 메모리 부족 강제종료, 프리웨어
OSForensics (PassMark)	Dirty상태 분석 X, 삭제레코드 분석 X 메모리 부족 강제종료 안됨, 유료
WinSearchDBAnalyzer (http://moaistory.blogspot.com)	Dirty상태 분석 O, 삭제레코드 분석 O 메모리 부족 강제종료 안됨, 프리웨어

[표 8. 색인파일 분석 프로그램 종류]

실험을 위하여 다음 [표9]와 같이 다양한 종류의 실험대상 외부저장장치를 준비하였다. 결과적으로 외장저장장치 종류나 파티션의 종류에 상관없이 색인정보를 구성하는 것이 가능했고 상황에 따라 색인정보가 수정되는 과정에서도 차이가 없었다. 색인정보에 볼륨시리얼이나 장치자체의 시리얼번호는 저장되지 않았고, 볼륨 레이블 정보는 저장되는 것을 확인하였다.

순번	제조사	종류	파티션	용량	볼륨 레이블	볼륨 시리얼	시리얼번호
1	LG Xtick (Innostor)	USB	FAT32 (M:)	32G	파티션0117	96f9-78e8	09020000000000000000000664
2	SanDisk (SanDisk)	USB	NTFS (L:)	64G	Sandk787	b012-4888	4C530001160320115062
3	LG Xtick (Innostor)	USB	exFAT (K:)	32G	설정안함	b453-77d9	09020000000000000000000715
4	삼성 (Portable_SSD_T5)	SSD	NTFS (G:)	1T	SST51091	3e69-cc0b	1234567A21C3
5	Seagate (Seagate)	HDD	NTFS (I:)	1T	SGT770	b89a-8a11	NA7S07TR
			NTFS (J:)		SGT990	067c-c338	
6	삼성	micro SD	exFAT (F:)	64G	MicSD555	18b8-6d3c	-

[표 9. 실험 대상 외부저장장치]

(1) 로컬PC 자료 → 외장저장장치 복사

실험 PC 바탕화면에 “Forensics No.1” 폴더 생성 후 폴더 내에 "포렌식 실습1.xlsx" 파일을 생성하였다. 그리고 “Forensics No.1” 폴더를 각 외장저장 장치에 복사하였다. 작업 후 색인파일 내부의 시간정보를 아래 [표10]과 같이 확인하였다.

외장저장장치	색인정보				비고	
	수집	생성	수정	접근	복사시간	종료/재부팅
LG Xtick (Innostor)	11.05.21:06	11.05.21:06	11.05.21:04	11.05.12:00	11.05.21:06	11.05.21:30/ 11.05.21:35
SanDisk (SanDisk)	11.05.21:06	11.05.21:06	11.05.21:04	11.05.21:06	11.05.21:06	11.05.21:30/ 11.05.21:35
LG Xtick (Innostor)	11.05.21:13	11.05.21:13	11.05.21:04	11.05.21:13	11.05.21:13	11.05.21:30/ 11.05.21:35
삼성 (Portable_SSD_T5)	11.05.21:05	11.05.21:05	11.05.21:04	11.05.21:05	11.05.21:05	11.05.21:30/ 11.05.21:35
Seagate (Seagate)	11.05.21:13	11.05.21:13	11.05.21:04	11.05.21:13	11.05.21:13	11.05.21:30/ 11.05.21:35
	11.05.21:13	11.05.21:13	11.05.21:04	11.05.21:13	11.05.21:13	11.05.21:30/ 11.05.21:35
삼성 MicroSD	11.05.21:05	11.05.21:05	11.05.21:04	11.05.21:05	11.05.21:05	11.05.21:30/ 11.05.21:35

[표 10. 로컬PC→외장저장장치 자료 복사 시 색인정보 확인]

색인파일 내부에 저장되는 생성, 수정, 접근 시간정보는 파일시스템의 MAC타임 정보와 동일하였다. 색인파일 내부에는 생성, 수정, 접근 시간 외에 색인정보가 수집된 시간(gatherTime)을 표시하는 필드가 존재하였다. 파일을 외장저장장치로 복사 하였을 때의 수집시간은 파일이 외장저장장치에 복사된 시간, 즉 해당 외장저장장치의 파일시스템에 생성된 시간으로 표시되었다.

(2) 외장저장장치 자료 수정

실험 PC 바탕화면에 “Forensics No.2” 폴더를 생성한 후 폴더 내에 ①“외장포렌식1.xlsx”, ②“외장포렌식2.xlsx” 2개의 파일을 생성하였다. “Forensics No.2” 폴더를 각 외장저장장치에 복사한 후 ①번 파일만 자료를 수정하고 시간 값 변동을 확인하였다. 그 결과는 [표11]과 같다.

외장저장장치	파일 구분	외장저장장치 시간정보				비고	
		수집	생성	수정	접근	복사/수정	종료/재부팅
LG Xtick (Innostor)	①	11.05.23:18	11.05.23:13	11.05.23:18	11.05.12:00	11.05.23:14/ 11.05.23:18	11.05.23:26/ 11.05.23:37
	②	11.05.23:14	11.05.23:14	11.05.23:11	11.05.12:00	11.05.23:14	11.05.23:26/ 11.05.23:37
SanDisk (SanDisk)	①	11.05.23:20	11.05.23:14	11.05.23:20	11.05.23:20	11.05.23:14/ 11.05.23:20	11.05.23:26/ 11.05.23:37
	②	11.05.23:14	11.05.23:14	11.05.23:11	11.05.23:14	11.05.23:14	11.05.23:26/ 11.05.23:37
LG Xtick (Innostor)	①	11.05.23:22	11.05.23:14	11.05.23:22	11.05.23:22	11.05.23:14/ 11.05.23:22	11.05.23:26/ 11.05.23:37
	②	11.05.23:14	11.05.23:14	11.05.23:11	11.05.23:14	11.05.23:14	11.05.23:26/ 11.05.23:37
삼성 (Portable_ SSD_T5)	①	11.05.23:22	11.05.23:14	11.05.23:22	11.05.23:22	11.05.23:14/ 11.05.23:22	11.05.23:26/ 11.05.23:37
	②	11.05.23:14	11.05.23:14	11.05.23:11	11.05.23:14	11.05.23:14	11.05.23:26/ 11.05.23:37
Seagate (Seagate)	①	11.05.23:21	11.05.23:14	11.05.23:21	11.05.23:21	11.05.23:14/ 11.05.23:21	11.05.23:26/ 11.05.23:37
		11.05.23:21	11.05.23:14	11.05.23:21	11.05.23:21	11.05.23:14/ 11.05.23:21	11.05.23:26/ 11.05.23:37
	②	11.05.23:14	11.05.23:14	11.05.23:11	11.05.23:14	11.05.23:14	11.05.23:26/ 11.05.23:37
		11.05.23:14	11.05.23:14	11.05.23:11	11.05.23:14	11.05.23:14	11.05.23:26/ 11.05.23:37
삼성 MicroSD	①	11.05.23:19	11.05.23:14	11.05.23:19	11.05.23:19	11.05.23:14/ 11.05.23:19	11.05.23:26/ 11.05.23:37
	②	11.05.23:14	11.05.23:14	11.05.23:14	11.05.23:14	11.05.23:14	11.05.23:26/ 11.05.23:37

[표 11. 외장저장장치 자료 수정 시 색인정보 확인]

실험 결과 파일을 외장저장장치에 복사하였을 때는 복사한(외장장치에 파일이 생성된) 시간을 수집시간으로 기록 하였고, 이후 외장저장장치에 저장된 파일이 수정되면 수정시간을 다시 수집시간으로 기록 한다는 것을 확인하였다. 외장저장장치에 저장된 파일에 대한 색인파일의 수집시간의 의미는 외장저장장치에 저장된 파일의 최근 변동사항 기록이라고 볼 수 있다.

(3) 외장저장장치에 저장된 파일 수정 없이 재부팅

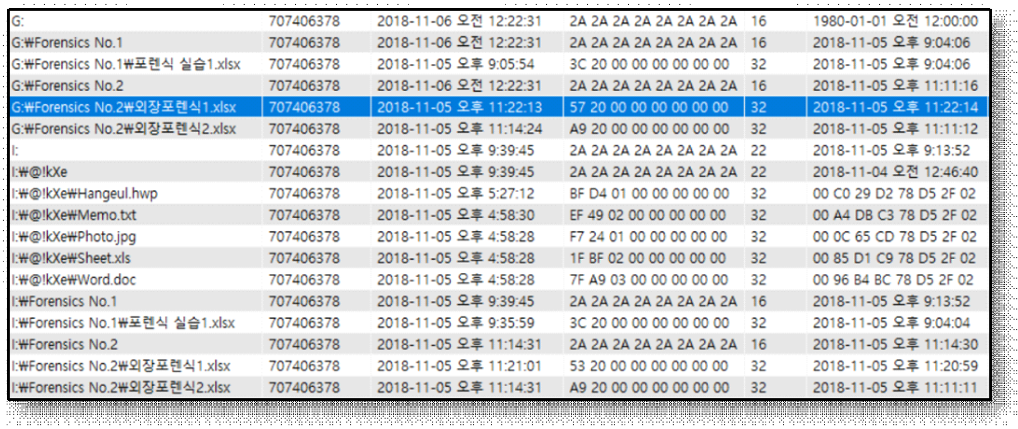
외장저장장치를 연결한 상태에서 외장저장장치에 저장된 파일에 대한 수정사항 없이 PC 작업을 진행하고 수회 재부팅 작업을 수행하였다. 그리고 색인파일 내부 데이터를 확인한 결과 외장저장장치와 관련한 필드의 시간정보에는 변동이 없었다. 외장저장장치가 연결되어 있더라도 저장된 파일에 특정 작업이 이루어지지 않는다면 시간정보가 갱신되지 않는다는 것을 알 수 있다.

(4) 외장저장장치 연결 해제 후 재부팅

외장저장장치를 연결 해제한 상태에서 수회 재부팅 하고 실험PC에서 다른 작업을 수행하였다. 그리고 색인파일 내부데이터를 확인한 바 외부저장장치의 색인내용이 변동 없이 그대로 남아 있는 것을 확인하였다. 결국 색인정보를 초기화하지 않는다면 외부저장장치의 색인정보는 색인파일 내부에 계속 남아 있는 것이다.

(5) 외장저장장치에서 자료 삭제

외장저장장치에 저장된 “Forensics No.2” 폴더 안의 ①″외장포렌식1.xlsx″파일을 일반삭제(delete)한 후 색인파일 내부 정보가 어떻게 변하는지 확인하였다.

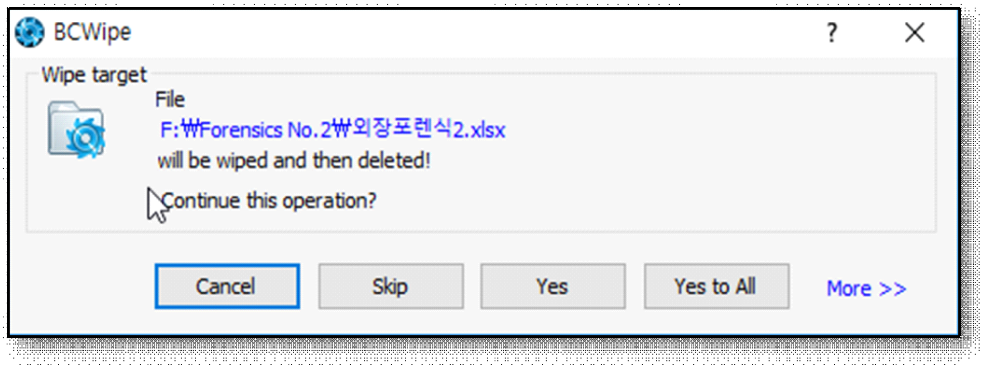


G:	707406378	2018-11-06 오전 12:22:31	2A 2A 2A 2A 2A 2A 2A 2A	16	1980-01-01 오전 12:00:00
G:₩Forensics No.1	707406378	2018-11-06 오전 12:22:31	2A 2A 2A 2A 2A 2A 2A 2A	16	2018-11-05 오후 9:04:06
G:₩Forensics No.1₩포렌식 실습1.xlsx	707406378	2018-11-05 오후 9:05:54	3C 20 00 00 00 00 00 00	32	2018-11-05 오후 9:04:06
G:₩Forensics No.2	707406378	2018-11-06 오전 12:22:31	2A 2A 2A 2A 2A 2A 2A 2A	16	2018-11-05 오후 11:11:16
G:₩Forensics No.2₩외장포렌식1.xlsx	707406378	2018-11-05 오후 11:22:13	57 20 00 00 00 00 00 00	32	2018-11-05 오후 11:22:14
G:₩Forensics No.2₩외장포렌식2.xlsx	707406378	2018-11-05 오후 11:14:24	A9 20 00 00 00 00 00 00	32	2018-11-05 오후 11:11:12
I:	707406378	2018-11-05 오후 9:39:45	2A 2A 2A 2A 2A 2A 2A 2A	22	2018-11-05 오후 9:13:52
I:₩@!kXe	707406378	2018-11-05 오후 9:39:45	2A 2A 2A 2A 2A 2A 2A 2A	22	2018-11-04 오전 12:46:40
I:₩@!kXe₩Hangeul.hwp	707406378	2018-11-05 오후 5:27:12	BF D4 01 00 00 00 00 00	32	00 C0 29 D2 78 D5 2F 02
I:₩@!kXe₩Memo.txt	707406378	2018-11-05 오후 4:58:30	EF 49 02 00 00 00 00 00	32	00 A4 DB C3 78 D5 2F 02
I:₩@!kXe₩Photo.jpg	707406378	2018-11-05 오후 4:58:28	F7 24 01 00 00 00 00 00	32	00 0C 65 CD 78 D5 2F 02
I:₩@!kXe₩Sheet.xls	707406378	2018-11-05 오후 4:58:28	1F BF 02 00 00 00 00 00	32	00 85 D1 C9 78 D5 2F 02
I:₩@!kXe₩Word.doc	707406378	2018-11-05 오후 4:58:28	7F A9 03 00 00 00 00 00	32	00 96 B4 BC 78 D5 2F 02
I:₩Forensics No.1	707406378	2018-11-05 오후 9:39:45	2A 2A 2A 2A 2A 2A 2A 2A	16	2018-11-05 오후 9:13:52
I:₩Forensics No.1₩포렌식 실습1.xlsx	707406378	2018-11-05 오후 9:35:59	3C 20 00 00 00 00 00 00	32	2018-11-05 오후 9:04:04
I:₩Forensics No.2	707406378	2018-11-05 오후 11:14:31	2A 2A 2A 2A 2A 2A 2A 2A	16	2018-11-05 오후 11:14:30
I:₩Forensics No.2₩외장포렌식1.xlsx	707406378	2018-11-05 오후 11:21:01	53 20 00 00 00 00 00 00	32	2018-11-05 오후 11:20:59
I:₩Forensics No.2₩외장포렌식2.xlsx	707406378	2018-11-05 오후 11:14:31	A9 20 00 00 00 00 00 00	32	2018-11-05 오후 11:11:11

[그림 13. 자료 삭제 후 색인파일 데이터 확인]

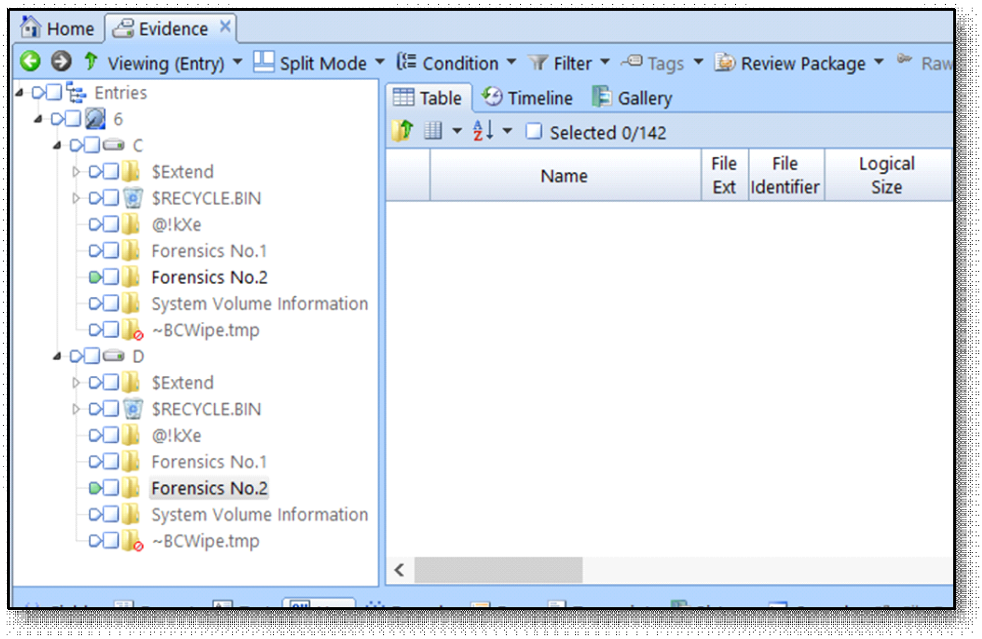
처음 재부팅 하였을 때는 [그림13] 과 같이 색인파일 내에는 삭제된 폴더와 파일의 정보가 업데이트 되지 않고 그대로 남아있는 것을 확인하였다. 수집 시간도 변동되지 않은 상태로 유지하였다. 하지만 이후 색인파일이 업데이트 되면서 해당 필드가 삭제되어 뷰어 프로그램에서도 해당 정보를 확인할 수 없었다.

두 번째로 외장저장장치에 저장된 “Forensics No.2” 폴더 내 ②“외장포렌식2.xlsx” 파일을 BCwipe[16]를 이용하여 완전삭제(와이핑) 하였다.

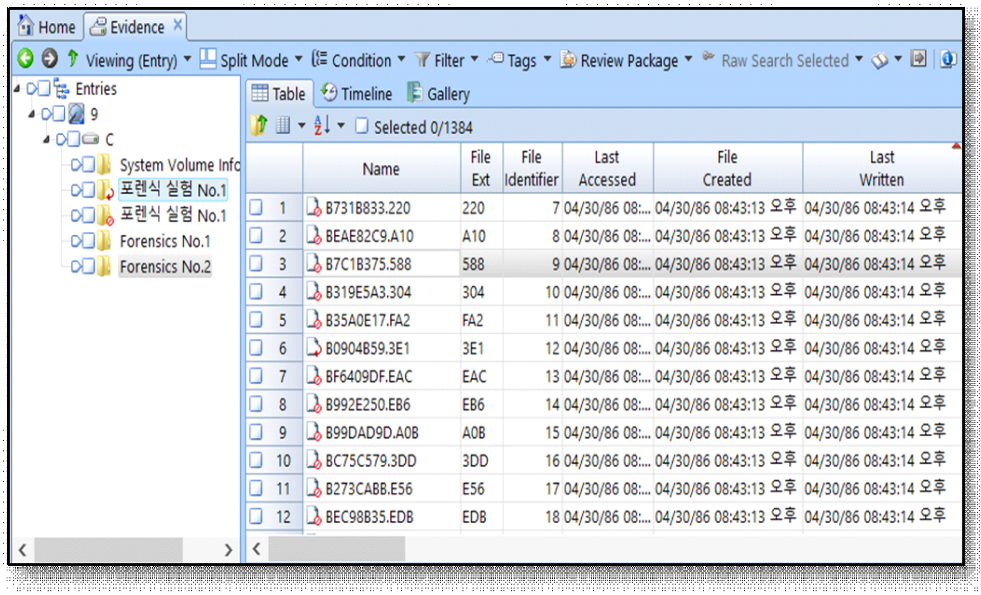


[그림 14. BCwipe를 이용한 파일 삭제]

16) jetico에서 개발한 파일이나 폴더를 복구가 불가능하도록 완전히 삭제해 주는 와이핑 프로그램.



[그림 15. NTFS 파일시스템에서 와이핑 확인]



[그림 16. FAT 파일시스템에서 와이핑 확인]

그리고 위의 [그림15] 와 [그림16] 과 같이 Encase(ver.7.12)[17]를 이용하여 와이핑 작업된 외부저장장치의 데이터를 확인하였다. 파일시스템에 따라서 파일 완전삭제 시 나타나는 결과가 다른 것을 볼 수 있다. NTFS 기반의 외부저장장치의 경우 "~BCwipe.tmp" 폴더 내에 LOGFILEWIPER 라는 로그파일이 남겨진 것을 확인할 수 있었고, FAT 기반의 외부저장장치의 경우는 폴더 내에 임의의 데이터들로 채워진 것을 확인 할 수 있었다. 삭제된 “외장포렌식2.xlsx” 파일의 흔적은 전혀 찾을 수 없었다.

2018-11-05 오후 9:39:45	2018-11-05 오후 9:13:49	2018-11-05 오후 9:13:49	2018-11-05 오후 9:13:51	J:₩Forensics No.1
2018-11-05 오후 9:36:07	2018-11-05 오후 9:04:04	2018-11-05 오후 9:13:49	2018-11-05 오후 9:13:51	J:₩Forensics No.1₩포렌식 실습1.xlsx
2018-11-05 오후 11:14:29	2018-11-05 오후 11:14:28	2018-11-05 오후 11:14:28	2018-11-05 오후 11:14:28	J:₩Forensics No.2
2018-11-05 오후 11:14:29	2018-11-05 오후 11:11:11	2018-11-05 오후 11:14:28	2018-11-05 오후 11:14:28	J:₩Forensics No.2₩외장포렌식2.xlsx
2018-11-06 오전 1:47:23	1980-01-01 오전 12:00:00	1980-01-01 오전 12:00:00	1980-01-01 오전 12:00:00	K:
2018-11-06 오전 1:47:22	2018-11-05 오후 9:04:06	2018-11-05 오후 9:13:04	2018-11-05 오후 9:13:04	K:₩Forensics No.1
1938-03-01 오후 1:16:23	2018-11-05 오후 9:04:06	2018-11-05 오후 9:13:04	2018-11-05 오후 9:13:04	K:₩Forensics No.1₩포렌식 실습1.xlsx
2018-11-06 오전 1:47:22	2018-11-05 오후 11:11:16	2018-11-05 오후 11:14:03	2018-11-05 오후 11:14:02	K:₩Forensics No.2
2018-11-05 오후 11:14:03	2018-11-05 오후 11:11:12	2018-11-05 오후 11:14:03	2018-11-05 오후 11:14:02	K:₩Forensics No.2₩외장포렌식2.xlsx
2018-11-05 오후 5:01:27	2018-11-05 오후 4:40:12	2018-11-04 오후 9:36:50	2018-11-05 오후 4:58:27	L:
2018-11-05 오후 9:35:55	2018-11-05 오후 9:06:48	2018-11-05 오후 9:06:48	2018-11-05 오후 9:15:08	L:₩Forensics No.1
2018-11-05 오후 9:06:49	2018-11-05 오후 9:04:04	2018-11-05 오후 9:06:48	2018-11-05 오후 9:06:48	L:₩Forensics No.1₩포렌식 실습1.xlsx
2018-11-05 오후 11:14:34	2018-11-05 오후 11:14:33	2018-11-05 오후 11:14:33	2018-11-05 오후 11:14:33	L:₩Forensics No.2
2018-11-05 오후 11:14:34	2018-11-05 오후 11:11:11	2018-11-05 오후 11:14:33	2018-11-05 오후 11:14:33	L:₩Forensics No.2₩외장포렌식2.xlsx
2018-11-06 오전 1:47:23	1980-01-01 오전 12:00:00	1980-01-01 오전 12:00:00	1980-01-01 오전 12:00:00	M:
2018-11-06 오전 1:47:22	2018-11-05 오후 9:04:06	2018-11-05 오후 9:06:11	2018-11-05 오전 12:00:00	M:₩Forensics No.1
2018-11-05 오후 9:06:12	2018-11-05 오후 9:04:06	2018-11-05 오후 9:06:11	2018-11-05 오전 12:00:00	M:₩Forensics No.1₩포렌식 실습1.xlsx
2018-11-06 오전 1:47:22	2018-11-05 오후 11:11:16	2018-11-05 오후 11:13:55	2018-11-05 오전 12:00:00	M:₩Forensics No.2
2018-11-05 오후 11:13:55	2018-11-05 오후 11:11:12	2018-11-05 오후 11:13:55	2018-11-05 오전 12:00:00	M:₩Forensics No.2₩외장포렌식2.xlsx

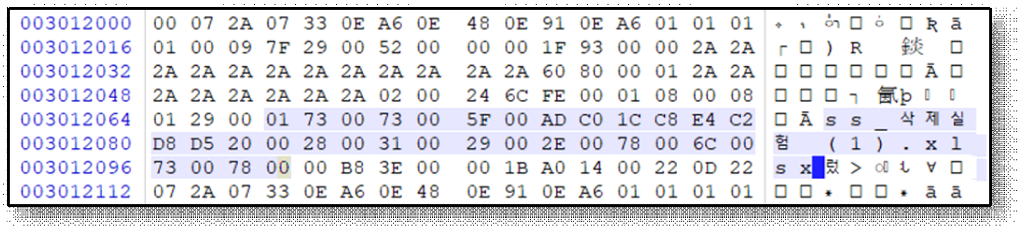
[그림 17. 완전삭제 후 색인파일 확인]

완전삭제의 경우도 [그림17]과 같이 처음 재부팅 후 색인파일 내에 삭제된 폴더와 파일의 정보가 그대로 유지되는 것을 확인할 수 있다. 하지만 일반삭제와 마찬가지로 이후 재부팅 과정을 거치면서 색인파일 내부 데이터도 업데이트가 되었다. 해당 레코드는 삭제되었고 EDB 뷰어 프로그램에서도 해당 데이터를 보여주지 못했다.

17) Guidance Software 사에서 개발한 디지털포렌식 분석 도구이다.

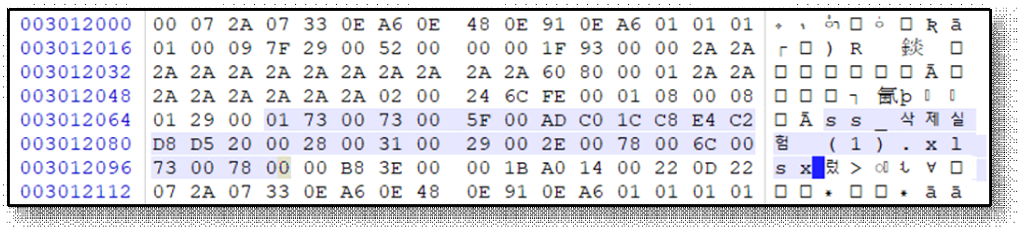
(6) 자료 삭제 시 오프셋 정보 확인

외장저장장치에서 자료를 삭제하였을 경우 색인파일에서 대상 정보가 실제로 삭제되는지 아니면 해당 레코드만 삭제 처리되고 정보는 계속 유지되는지 실험을 하였다. 아래의 [그림18]은 “ss_삭제실험(1)” 이란 파일을 외장저장장치에 저장한 후 X-Ways Forensics[18]를 이용하여 색인파일 내부에서 해당 오프셋 정보를 확인한 것이다.



[그림 18. 파일생성 후 해당 오프셋 확인]

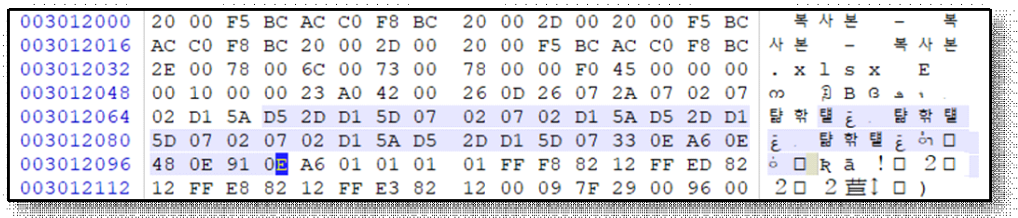
그리고 “ss_삭제실험(1)” 이란 파일을 외장저장장치에서 삭제하였다. 이후 몇 번의 재부팅 과정을 거치면서 색인파일이 업데이트 되었고 EDB 뷰어프로그램에서는 해당 레코드가 삭제되어 보이지 않았다. 하지만 해당 오프셋 정보를 확인하니 [그림19]와 같이 데이터가 삭제되지 않고 유지되는 것을 확인하였다.



[그림 19. 파일삭제 후 해당 오프셋 확인]

18) 독일 X-Ways Software Technology AG사의 대표적인 컴퓨터 포렌식 소프트웨어.

그리고 해당 외장저장장치에 다른 파일을 계속 복사하는 실험을 하였다. 해당 오프셋을 확인하니 [그림20]과 같이 외장저장장치에 새로 복사된 파일에 관한 데이터로 덮여 쓰여진 것을 확인하였다.



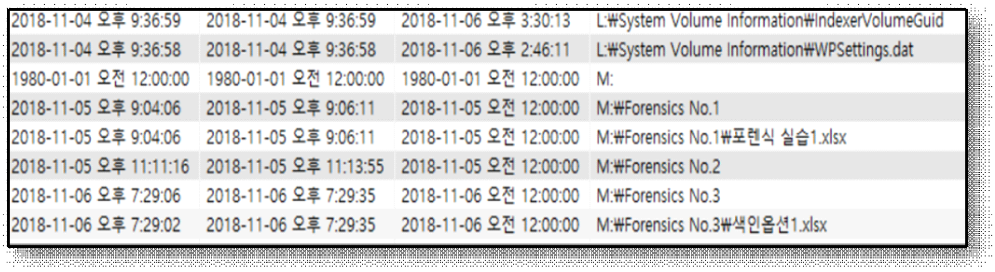
[그림 20. 파일복사 후 해당 오프셋 확인]

(7) 외장저장장치 포맷 레이블 변경

색인대상인 USB(볼륨레이블:Sandisk787)를 포맷하고 볼륨 레이블을 다른 이름(Sandisk890)으로 지정하였다. 동일한 USB이지만 볼륨 레이블이 새로 지정되었을 경우 이 새로운 볼륨은 색인 대상에서 자동으로 제외되는 것을 확인하였다. 하지만 이전에 색인되었던 정보는 유지된다.

(8) 색인대상에서 외장저장장치 제외

외장저장장치 중 (볼륨레이블-파티션0117)에 "Forensics No3" 폴더를 생성하고 "색인옵션1.xlsx" 파일을 저장하였다. 그리고 색인파일에 관련 내용이 업데이트된 것을 확인하였다. 그리고 제어판의 색인옵션을 통해서 해당 외장저장장치를 색인대상에서 제외하였다. 하지만 [그림21]과 같이 색인대상에서 제외된 외장저장장치의 색인정보는 지워지지 않고 계속 유지되는 것을 알 수 있다.



2018-11-04 오후 9:36:59	2018-11-04 오후 9:36:59	2018-11-06 오후 3:30:13	L:₩System Volume Information₩IndexerVolumeGuid
2018-11-04 오후 9:36:58	2018-11-04 오후 9:36:58	2018-11-06 오후 2:46:11	L:₩System Volume Information₩WPSettings.dat
1980-01-01 오전 12:00:00	1980-01-01 오전 12:00:00	1980-01-01 오전 12:00:00	M:
2018-11-05 오후 9:04:06	2018-11-05 오후 9:06:11	2018-11-05 오전 12:00:00	M:₩Forensics No.1
2018-11-05 오후 9:04:06	2018-11-05 오후 9:06:11	2018-11-05 오전 12:00:00	M:₩Forensics No.1₩포렌식 실습1.xlsx
2018-11-05 오후 11:11:16	2018-11-05 오후 11:13:55	2018-11-05 오전 12:00:00	M:₩Forensics No.2
2018-11-06 오후 7:29:06	2018-11-06 오후 7:29:35	2018-11-06 오전 12:00:00	M:₩Forensics No.3
2018-11-06 오후 7:29:02	2018-11-06 오후 7:29:35	2018-11-06 오전 12:00:00	M:₩Forensics No.3₩색인옵션1.xlsx

[그림 21. 색인제외 후 색인파일 확인]

(9) 다른PC에서 외장저장기기에 자료 복사한 후 실험PC 접속

색인정보의 수집시간은 실험PC에 접속된 시간으로 기록되고, 생성-수정-접근 시간은 다른PC로부터 생성된 MAC타임 정보를 기록하였다.

(10) 로컬PC 자료 수정

실험 PC 바탕화면에 “포렌식 회의록” 이란 폴더를 만들고 “회의록_1차.xlsx”파일을 저장했다. 그리고 시간대 별로 나누어서 저장 및 수정을 하였다. 그 결과 다음의 [표12] 와 같은 데이터를 수집하였다. 외장저장장치와 달리 로컬 PC에 저장된 데이터는 색인파일 내부 수집 시간이 윈도우 종료시간과 거의 유사하게 기록된다. 각 데이터의 수정내용이 temp 파일 형태로 저장되었다가 PC를 종료하면서 색인 파일에 반영되고 접근시간이 갱신되기 때문인 것으로 추정된다.

파일명	색인파일 시간정보				윈도우 종료
	수집	생성	수정	접근	
회의록_1차	11.06.23:27	11.06.20:05	11.06.20:05	11.06.23:27	11.06.23:29
회의록_2차	11.06.23:28	11.06.20:12	11.06.23:28	11.06.23:28	11.06.23:29
회의록_3차	11.06.23:35	11.06.20:17	11.06.20:17	11.06.23:35	11.06.23:36
회의록_4차	11.06.23:40	11.06.20:27	11.06.20:27	11.06.23:40	11.06.23:40
회의록_5차	11.06.23:50	11.06.20:33	11.06.20:33	11.06.23:50	11.06.23:52
회의록_6차	11.06.23:51	11.06.20:57	11.06.23:51	11.06.23:51	11.06.23:52
회의록_7차	11.07.00:04	11.06.23:02	11.06.23:02	11.07.00:04	11.07.00:06

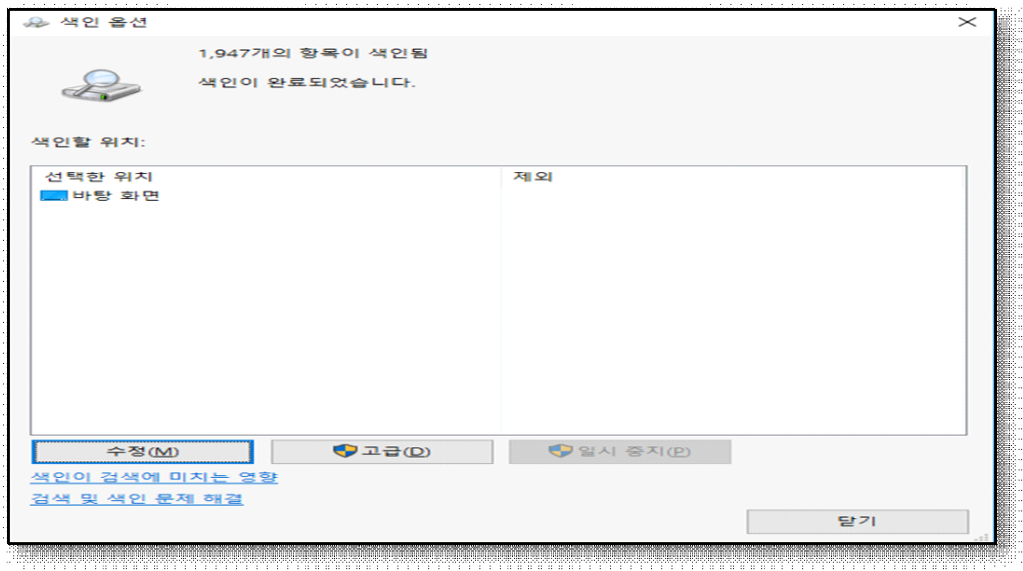
[표 12. 로컬PC 자료 수정 시 색인정보 확인]

5.2. 윈도우 색인파일 활용 방안

앞장에서 실험을 통해 확인한 색인정보의 특성을 활용하여 디지털포렌식 현장수사에 활용하는 방안을 연구해 보도록 하겠다.

(1) 외장저장장치 로그로서 활용

사용자가 외장저장장치에 대하여 단 한번이라도 색인을 걸었다면 그 색인정보는 사용자 PC의 특정경로에 Windows.edb 파일로 저장된다. 사용자가 외장저장장치의 연결을 해제한 경우에도 색인정보를 초기화 하지 않는 이상 이 색인정보는 계속 유지되는 특성이 있다. 외장저장장치의 연결이 해제된 경우 윈도우 색인옵션에서는 [그림22]와 같이 대상 기기의 볼륨이 사라지기 때문에 외장장치에 대하여 색인이 걸려있는지 여부는 실제 색인파일 내부 정보를 확인해야 알 수 있다.

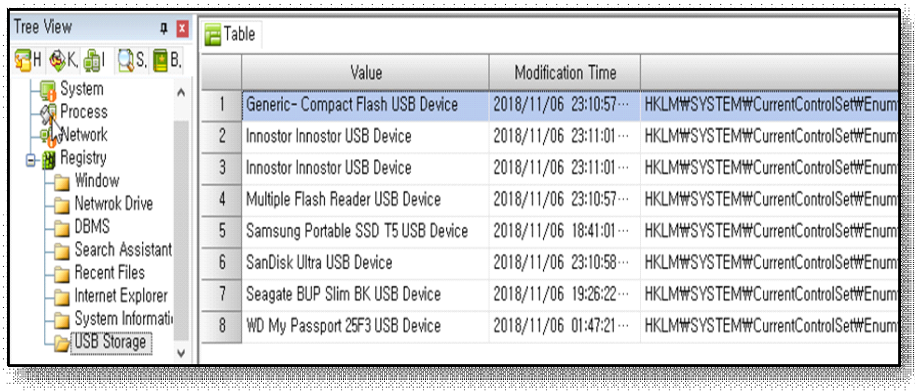


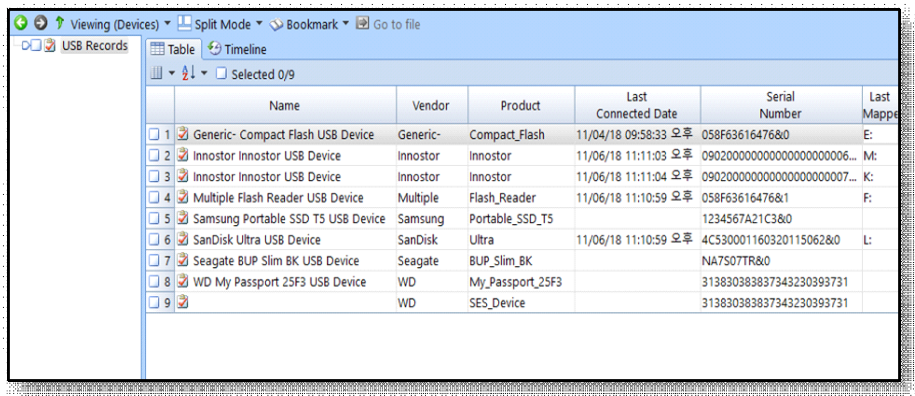
[그림 22. 외장저장장치 연결 해제-색인옵션]

기존의 포렌식 툴에서 확인할 수 있는 외장저장장치 로그 정보는 레지스트리 정보를 기반으로 분석한 외장저장장치의 제조사, 제품명, 접속시간, 시리얼 넘버 등이었고, 이 정보는 기기 자체에 대한 로그 정보였다. 사용자가 외장저장장치에 저장된 파일을 실행한 경우에 생성되는 링크 파일 분석 등의 경우를 제외하고는 외장저장장치 내부파일에 대한 정보는 확인할 수 없다.

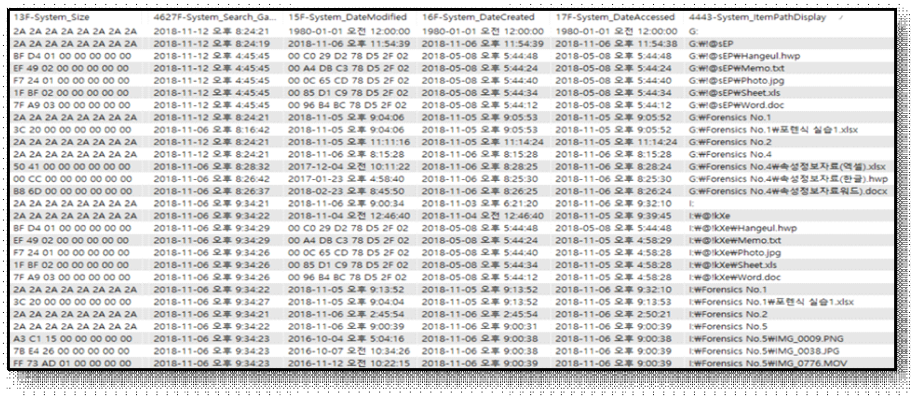
하지만 사용자가 만약 외장저장장치에 대하여 색인을 걸었고, 포렌식 수사관이 색인파일을 현장에서 분석한다면 외장저장장치에 저장된 각 파일명과 파일사이즈, 시간정보, 경로와 해당 볼륨레이블 정보까지 확인할 수 있다. 따라서 중요파일이 어떤 외장저장장치에 저장되었는지 파악하고 이 기기의 제출을 피압수자에게 요구할 수 있다. 아래의 [표13]은 포렌식 툴에서 확인한 로그정보와 색인정보에서 확인한 로그정보를 비교한 것이다. 색인정보의 크기는 제한이 없이 계속 증가하기 때문에 수많은 데이터가 저장될 수 있고 강력한 로그가 될 수 있다.

포렌식 툴 분석(매체 자체 로그)





색인정보 분석(파일 로그)



[표 13. 외장저장장치 로그 비교]

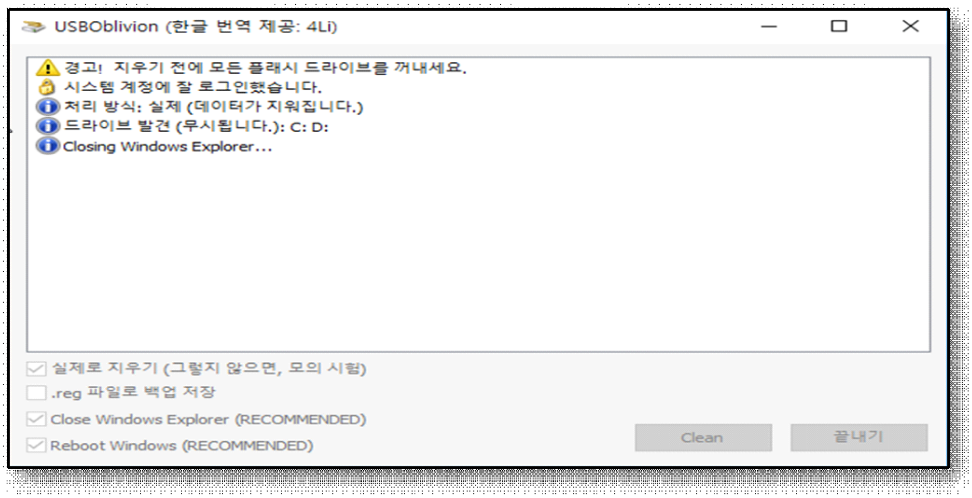
또한 기술유출 사건의 경우는 이 색인정보에 기록된 레코드 정보 자체가 아주 중요한 증거가 될 수 있다. 색인정보의 특성상 반드시 외장저장장치가 사용자 PC에 접속되었을 경우만 색인정보가 생성-기록되기 때문에 만약 색인정보에 기술유출 사건의 대상이 되는 중요 파일이 기록되어 있다면 이것으로 혐의자가 중요파일이 저장된 외장저장장치를 대상 PC에 연결하였다는 것을 입증할 수 있다.

(2) 외장저장장치 접속기록 삭제된 경우 활용

사용자가 의도적으로 외장저장장치에 대한 접속기록을 지웠을 경우에도 이를 확인할 수 있는 로그로 사용할 수 있다. 윈도우에서 외장저장장치의 접속 기록이 저장되는 곳은 앞장에서 설명하였듯이 레지스트리와 Setupapi.dev.log 그리고 이벤트로그 파일이다. Setupapi.dev.log 파일은 텍스트 파일로서 쉽게 삭제가 가능하고, 이벤트로그 파일 역시 이벤트뷰어에서 로그 지우기로 쉽게 삭제가 가능하다.

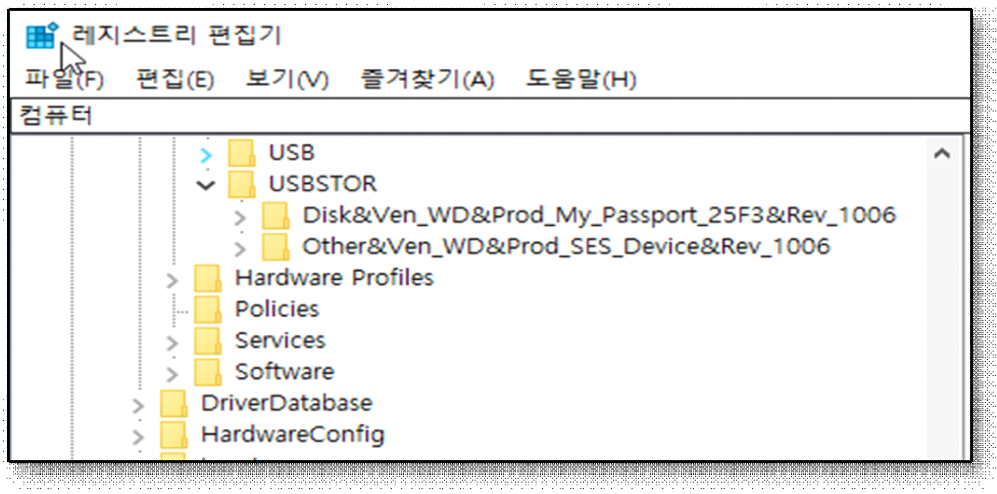
레지스트리의 경우는 관리자권한을 얻은 후 레지스트리 편집기에서 외장저장장치와 관련된 각각의 키를 직접 삭제하는 방법도 있지만 USB 접속기록 삭제 프로그램을 이용하면 더 쉽게 삭제할 수 있다. 대표적인 프로그램 “USBOblivion”[19] 을 이용해서 [그림23]과 같이 외장저장장치 접속기록을 삭제한 후 포렌식 툴로 확인해 보도록 하겠다.

19) USB 디바이스, CD-ROM 등의 접속기록 삭제 프로그램. 프리웨어. https://www.cherubicsoft.com/en/projects/usboblivion

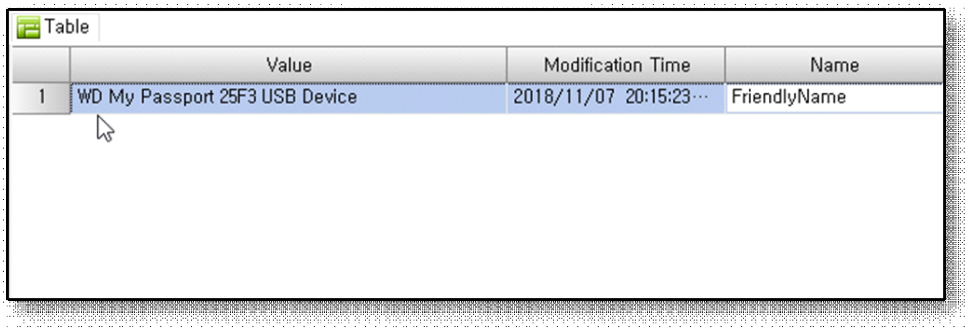


[그림 23. USBOblivion 이용하여 접속기록 삭제]

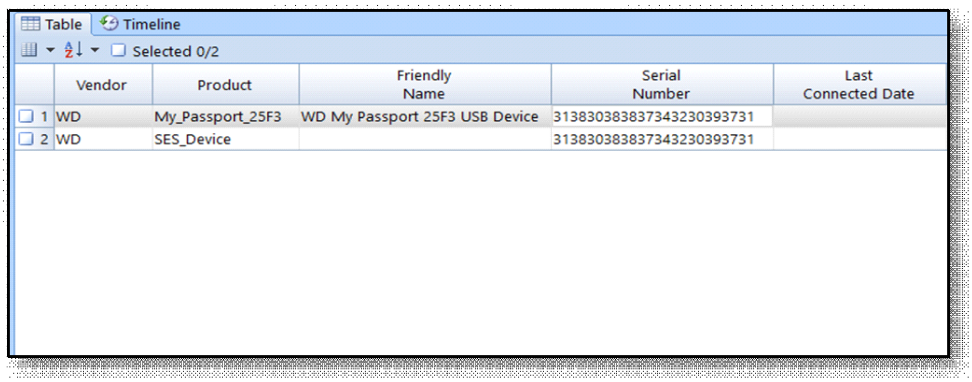
이 프로그램을 사용하여 접속기록을 삭제한 경우에는 [그림24]과 같이 외장저장장치 관련 레지스트리 정보가 모두 지워지고 [그림25]과 [그림26]에서 보는 것과 같이 포렌식툴 에서도 접속기록을 찾아낼 수 없었다.



[그림 24. 레지스트리에서 관련 데이터 삭제됨]



[그림 25. CFT에서 접속기록 확인불가]



[그림 26. Encase에서 접속기록 확인불가]

결과적으로 외장저장장치 접속기록을 완벽하게 제거해주는 프로그램을 이용하면 레지스트리 정보가 완전히 삭제되고, 포렌식 툴로도 흔적을 찾을 수 없었다. 포렌식 수사관은 점프리스트나 링크파일 분석을 통하여 외장저장장치의 흔적을 찾아보거나, 프리패치를 분석하여 접속기록을 지워주는 프로그램의 사용 흔적을 확인할 수 있다.

하지만 이런 경우에도 외장저장장치에 대한 색인이 구성되어 있다면 색인파일을 활용하여 외장저장장치의 접속기록과 내부 정보를 활용할 수 있다.

(3) 외장저장장치 접속시간 추정

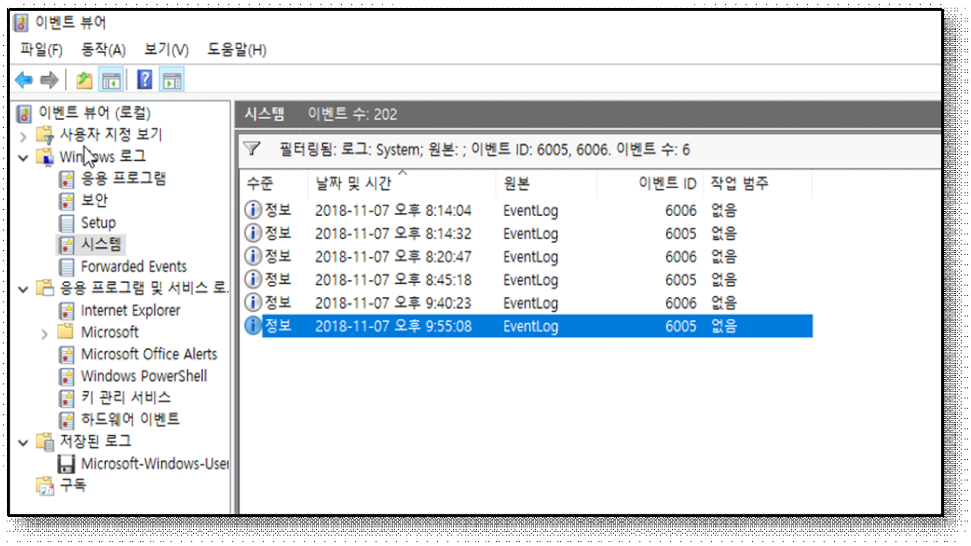
사용자의 PC에 외장저장장치가 언제 접속하였는지 확인하는 것이 압수수색에서 중요한 쟁점사안이 되는 경우가 있다. 포렌식 툴에서는 레지스트리 정보를 분석하여 외장저장장치의 접속시간을 확인한다. 레지스트리 정보는 USB디바이스의 최초 접속시점, 부팅 후 최초 접속시점, 마지막 접속 시점 등 제한적인 시간정보만 확인할 수 있다.

이런 경우에도 색인정보를 활용할 수 있다. 이전 장에서 살펴본 바와 같이 외장저장장치에 대한 색인정보는 해당 장치의 파일시스템에 파일이 처음 복사(생성)된 시점과 파일이 수정된 시점 등 변경사항이 있을 경우에 수집시간이 기록된다.

수집시간이 기록되었다는 의미는 외장저장장치를 해당 PC에 연결하였다는 의미가 된다. 이 수집시간을 시간 순으로 정렬하여 보면, 외장저장장치가 각각 어느 시점에 접속하였는지 확인할 수 있다. 결과적으로 레지스트리의 시간정보보다 훨씬 더 많은 시간정보를 획득할 수 있다.

(4) 사용자 PC 로그오프 시간 특정

사용자가 PC를 어느 시점에 사용하였는지 역시 수사에 중요한 단서가 될 수 있다. 사용자가 PC를 켜고 끈 시간은 윈도우 이벤트로그로 기록된다. 다음 [그림27]은 이벤트뷰어에서 관련 이벤트 항목을 확인한 것이다. [표14]는 사용자 계정 로그온/오프 와 관련 이벤트를 모아놓은 것인데 이 이벤트뷰어에서 이 항목을 필터링 하여 사용자가 언제 PC를 부팅하고 언제 종료하였는지 확인할 수 있다.



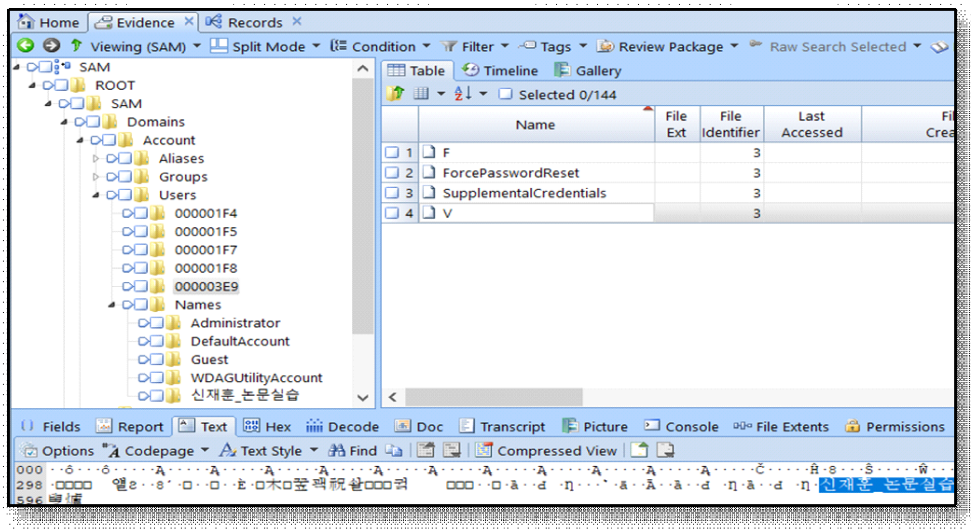
[그림 27. 이벤트뷰어에서 관련 이벤트 확인]

이벤트ID	설명
6005	이벤트 로그 서비스 시작(부팅시 기록)
6006	정상적으로 시스템 종료 시 기록
6008	비정상적인 시스템 종료 시 기록
6009	부팅 시 OS버전, 빌드 번호 등 정보 기록
4624	계정 로그인
4647	계정 로그아웃

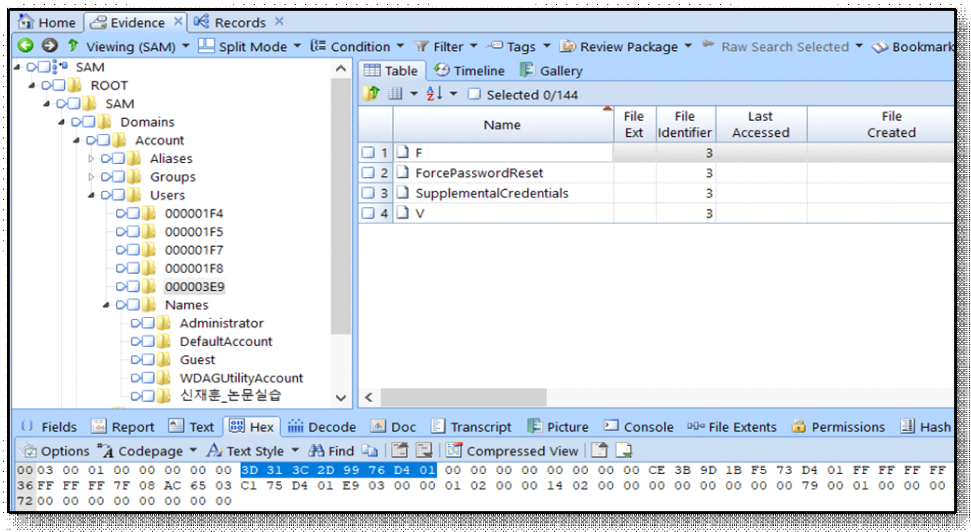
[표 14. 로그온/오프 관련 이벤트ID]

하지만 이런 이벤트로그는 윈도우 이벤트뷰어에서 로그지우기를 클릭하는 것만으로 아주 쉽게 삭제할 수 있다.

사용자가 PC에 로그온, 로그오프한 시간은 레지스트리에서도 확인할 수 있다. 먼저 사용자 계정을 생성한 시간은 C:\users\%user_name%\폴더 안에 있는 NTUSER.DAT 파일의 생성시간으로 확인할 수 있다. 이 파일의 마지막 쓰기 시간은 사용자 계정을 로그오프한 시간이다.



[그림 28. 레지스트리에서 계정명 확인]



[그림 29. 레지스트리에서 계정 로그온/로그오프 시간 확인]

위의 [그림28]과 [그림29]는 Encase를 이용하여 레지스트리에서 사용자 계정이름과 로그온/로그오프 시간을 확인한 내용이다. RID 아래 F값을 확인하면 상세시간 정보를 알 수 있다. 아래 [표15]은 각 오프셋에 시간정보가 어떻게 기록되는지 나타낸 표이다. 모든 로그오프 시간을 기록하는 것은 아닌 것을 알 수 있다.

오프셋	설명
8~15	마지막 계정 로그인 시간
24~31	마지막 패스워드 변경 시간
32~39	계정 만료 시간
40~47	마지막 로그인 실패 시간

[표 15. 로그온/오프 시간관련 오프셋 정보]

이벤트 로그파일이 삭제되었을 경우나 레지스트리 정보보다 더 자세한 정보를 얻고 싶을 때 색인정보를 활용할 수 있을 것이다. 만약 색인정보가 구성되어 있다면 이 정보를 이용하여 사용자의 로그오프 기록을 확인할 수 있다. 앞장에서 살펴보았듯이 사용자 PC에 저장된 파일의 변경이 있었을 경우 색인정보는 PC가 종료되면서 이런 정보들을 수집한다. 그리고 이런 정보는 색인파일에 각 파일별 수집시간으로 기록된다. 이런 수집시간은 색인대상으로 선택된 파티션이나 폴더에 변경사항이 있을 때에만 기록되기 때문에 이것으로 모든 로그오프 시간을 특정 지을 수는 없다. 하지만 사용자가 PC를 사용하면서 수집된 색인정보가 많을 경우 이 자료를 활용할 가치는 분명히 있을 것이다.

(5) 파일 완전삭제 시 흔적 분석

압수수색 현장에서 사용자가 PC의 중요 증거파일을 완전삭제(와이핑)하였을 경우 이 파일의 흔적이 색인파일 내부에 남아있을 가능성이 있다. 현장에서 $mft 파일과 색인 파일을 비교-분석하여 $mft에는 없는 정보가 만약 색인파일에 존재한다면 이 정보에 주목해야 할 필요가 있다. 이 파일이 $mft에서 삭제된 이유가 있는지 확인하고, 추가적으로 프리패치 분석 등을 통하여 실행 프로그램 중 완전삭제 프로그램의 흔적이 있는지 확인 하여야 한다. 색인파일은 PC 재부팅 과정을 통하여 업데이트 되므로 완전삭제 직후 압수현장에서 확인할 수 있는 방법이다.

결 론

본 논문은 윈도우에서 색인정보를 저장하는 Windows.edb 파일을 활용하여 외장저장장치의 흔적을 탐색하는 방법에 대하여 기술하였다. 심도 깊은 분석방법이나 새로운 기법을 제시한 것이 아니다. 단지 많은 사람들이 크게 관심을 두지 않는 색인파일을 분석하고 이 색인정보에서 외장저장매체의 흔적이 존재할 가능성을 제시 한 것이다.

외장저장장치의 흔적을 찾는 포렌식 분석기법들은 이미 널리 알려져 있다. 포렌식 프로그램은 레지스트리 정보 등을 분석하여 외장저장장치의 로그를 확인한다. 하지만 기존의 이런 로그정보들은 외장저장장치 기기 자체에 대한 정보일 뿐 외장저장장치 내부에 어떤 데이터가 저장 되었는지는 확인할 수 없다. 만약 사용자가 외장저장장치에 색인을 걸어 두었다면 색인파일 내부에 외장저장장치에 저장된 각 파일들의 파일명, 경로, 시간정보 등이 그대로 기록된다. 그렇기 때문에 색인정보는 아주 강력한 로그로 사용할 수 있다. 또한 볼륨레이블 정보가 기록되기 때문에 이를 토대로 중요한 파일이 담긴 대상매체를 확인하고 그 기기의 제출을 요구할 수도 있다.

색인정보를 활용하는 것은 분명한 한계가 있다. 사용자가 외장저장장치에 대한 색인을 허용하였을 경우에만 이 정보가 생성-유지되기 때문이다. PC를 사용하면서 외장저장장치에 대하여 사용자가 색인을 걸게 될 확률이 얼마나 될까? 정확한 통계를 낼 수는 없지만 그렇게 높지는 않을 것으로 예상된다. 하지만 분명한 것은 확률이 제로가 아니라는 것이다. 회사에서 담당자가 업무용 데이

터가 많이 저장된 외장하드를 사용하면서 탐색기의 검색 창에서 찾고자 하는 파일명을 한번쯤 검색 할 수 있다. 그리고 무심코 '색인에 추가' 버튼을 누를 수도 있다. 이렇게 본인도 모르는 사이에 색인에 추가된 드라이브나 폴더가 있을 가능성은 분명히 존재한다.

포렌식 수사관은 아주 작은 가능성도 놓치지 말아야 한다. 기존 외장저장장치 분석 아티팩트와 함께 색인파일을 추가적으로 분석한다면 현장에서 주요 증거물을 찾는데 분명히 도움이 될 수 있을 것이라 생각한다.

참 고 문 헌

<단행본>

1. 고원봉, “윈도우 포렌식 실전가이드”, 한빛미디어(2010)
2. Harlan Carvey, “Windows Registry Forensics”, SYNGRESS(2011)
3. Harlan Carvey, “Windows Forensic Analysis Toolkit", SYNGRESS (2012)
4. Pavel Yosifovich 외, “Windows Internals”, Microsoft Press(2017)

<논문>

1. JM. Gordon, “A forensic examination of windows desktop - 45 search (version 3)”, Master’s Thesis, Cranfield University, 2009
2. J. Douglas, “Forensic artefacts present in microsoft windows deskto p search,” Master’s Thesis, Cranfield University, 2009
3. H. Chivers and C. Hargreaves, "Forensic data recovery from the w indows search database", Digital Investigation, vol. 7, no. 3-4, p. 11 4-126, Apr. 2011.
4. H. Chivers, “Private browsing: A window of forensic opportunity”, Digital Investigation, vol. 11, no. 1, pp. 20-29, Mar. 2014.
5. Jeonghyeon Kim, "Digital Forensic Investigation of ESE Database" Master’s Thesis, Korea University, 2015
6. 김정현, 최종현, 이상진, "ESE 데이터베이스 내의 삭제된 레코드 복구 기법" , 정보보호학회논문지, 25(5), 1143-1151. Figure 1. 2015.

<인터넷 자료>

1. J. Metz, "Extensible Storage Engine Database File format specification",(https://github.com/libyal/libesedb/blob/master/documentation/Extensible%20Storage%20Engine%20(ESE)%20Database%20File%20(EDB)%20format.asciidoc)
2. NirSoft, “ESEDatabaseView v1.30”, (http://www.nirsoft.net/utils/ese_database_view.html)
3. 김진국, “윈도우에서 USB 흔적 추적하기”, (http://forensic-proof.com/archives/3632)
4. 김진국, “윈도우 7 장치 연결/해제 이벤트 로그 ”, (http://forensic-proof.com/archives/5945)

Abstract

External storage device trace analysis using Windows index file

Jaehun Shin

The Graduate School

Seoul National University

A digital forensic investigator faces a variety of situations in the process of enforcing search. Depending on how cool and fast the digital forensic investigator responds to these situations, the results of the seizure search will vary greatly.

At the recent seizure site, determining which external storage device was connected to the user 's PC and when that point became an important key point of the seizure search. Many people back up their work-related data to external hard drives or USB. Because of the high probability that important material related to the case is stored in this external storage device, the investigator should promptly determine which external storage device the user owns and secure the device at the seized search site.

Many forensic analysis techniques such as registry analysis and li nk file analysis are already known for finding traces of external stora ge devices. However, not many have noticed the Windows.edb file, w hich is a Windows index file. This index information is generated wh en the user indexes a specific volume or folder, and the information i s retained unless the index information is initialized. An external stor age device may also be included in the index object.

This index information can only be used in very limited cases beca use there is a prerequisite that the user has indexed. However, in ma ny cases, the user is unknowingly indexing a specific volume or folde r while using the PC. If you have indexed once, the index on the tar get volume will still be maintained, so there is a clear possibility that you will get important data that will be key to the event based on th is information. From the viewpoint of the investigator, one of the sm all possibilities can not be neglected. Information about this index file should also be checked.

In this paper, we will briefly review the structure of the Windows.e db file, which is a Windows index file, and then describe the convent ional forensic artifacts for analyzing the traces of external storage de vices. We will then experiment with how the data in this index file will be created and changed, and then we will characterize the index file. And we propose a method to utilize the index file based on this result.

keywords **:** *Windows.edb, Windows search, Index data, External stor age*

Student Number : 2017-21469