Advanced $UsnJrnl Forensics

blueangel

blueangel1275@gmail.com

http://forensic-note.blogspot.kr/

Junghoon Oh

1. $UsnJrnl

2. $UsnJrnl Record Carving

3. NTFS Log Tracker v1.4

4. Conclusion

$UsnJrnl

NTFS 변경 로그 파일

- **응용 프로그램이 특정 파일의 변경 여부를 파악하기 위해 사용**

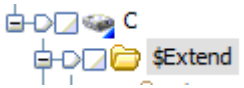
- **기본적으로 Windows 7 부터 활성화되어 있음**
  - 비활성화 되어있을 시, Fsutil 로 활성화 시킬 수 있음

    > fsutil usn [createjournal] m=<MaxSize> a=<AllocationDelta> <VolumePath>
  - Fsutil 의 자세한 사용법은 http://technet.microsoft.com/en-us/library/cc788042.aspx

- **$Max 속성과 $J 속성으로 구성**
  - $Max : 변경 로그의 기본 메타 데이터 저장
  - $J 속성 : 실제 변경 로그 레코드 저장
    - ✓ 각 레코드들은 USN(Update Sequence Number) 정보를 가짐
    - ✓ USN 정보를 통해 각 레코드들의 순서 구분
    - ✓ 실제 USN 값은 $J 속성 내에서의 레코드의 Offset 값
    - ✓ USN 값은 MFT 엔트리의 $STANDARD_INFORMATION 속성에도 저장되어 있음

NTFS 변경 로그 파일(계속)

- **위치**
  - 루트에 있는 "$Extend" 폴더 아래 위치



Name	File Created	Last Written	Entry Modified	Last Accessed	Logical Size
$UsnJrnl:$J					1,246,483,680
$UsnJrnl:$Max					32

- **기록 되는 로그 데이터의 양(일반적으로...)**
  - 컴퓨터를 계속 사용할 경우, 1~2일 정도의 로그가 남음
  - 규칙적으로 쓸 경우(하루 8시간), 4~5일 정도의 로그가 남음

- **포렌식 준비도**
  - 로그 용량을 크게 재설정

- **Digital Forensic Profit**
  - 특정 기간 내의 모든 파일 시스템 히스토리(생성, 삭제, 수정...) 기록을 알 수 있음~!!

$Max 속성의 구조

- **$Max 속성의 크기**
  - 32 Byte 고정 크기를 가짐

- **$Max 속성의 저장 정보**

Offset	Size	Stored Information	Detail
0x00	8	Maximum Size	로그 데이터의 최대 크기
0x08	8	Allocation Size	새로운 데이터가 저장될 때 할당 되는 영역의 크기
0x10	8	USN ID	"$UsnJrnl" 파일의 생성시간(FILETIME)
0x18	8	Lowest Valid USN	현재 저장된 레코드 중 가장 작은 USN 값 이 정보를 통해 $J 속성 내 첫 번째 레코드로 바로 이동 가능

$J 속성 구조

- **가변 크기의 로그 레코드들이 연속적으로 나열됨**

- **속성의 앞 부분은 0으로 채워진 "Sparse Area" 를 가짐**

- 이러한 구조를 가지는 이유는 운영체제가 $J 속성에 저장되는 로그 데이터의 크기를 일정하게 유지하려고 하기 때문임

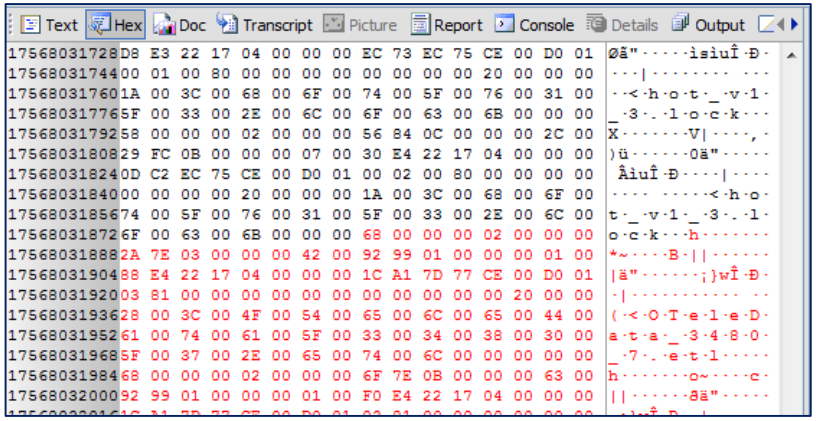
- $J 속성의 레코드 할당 정책
  1. 새로운 로그 레코드들은 속성 끝에 추가됨
  2. 추가된 레코드들의 총 크기가 "Allocation Size"를 넘으면 추가 레코드들을 포함하여 전체 로그 데이터의 크기가 "Maximum Size" 를 넘는지 확인
  3. 전체 로그 데이터의 크기가 "Maximum Size" 를 넘는 다면 로그 데이터의 앞 부분을 "Allocation Size" 만큼 0으로 채워 "Sparse Area" 로 만듬(실제 해당 디스크 영역을 0으로 채우는 것은 아님~!!)

- 따라서 $J 속성의 논리적인 크기는 계속 커지지만 실제 데이터가 할당된 영역은 일정하게 유지됨
- 일반적으로 0x2000000 ~ 0x23FFFFF 의 로그 데이터를 저장

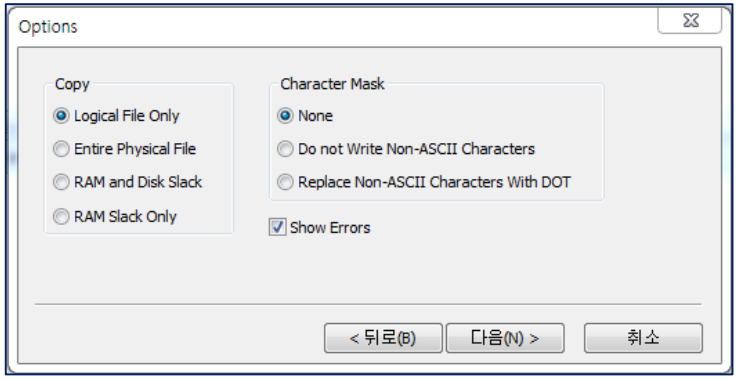
$J 속성 구조(계속)

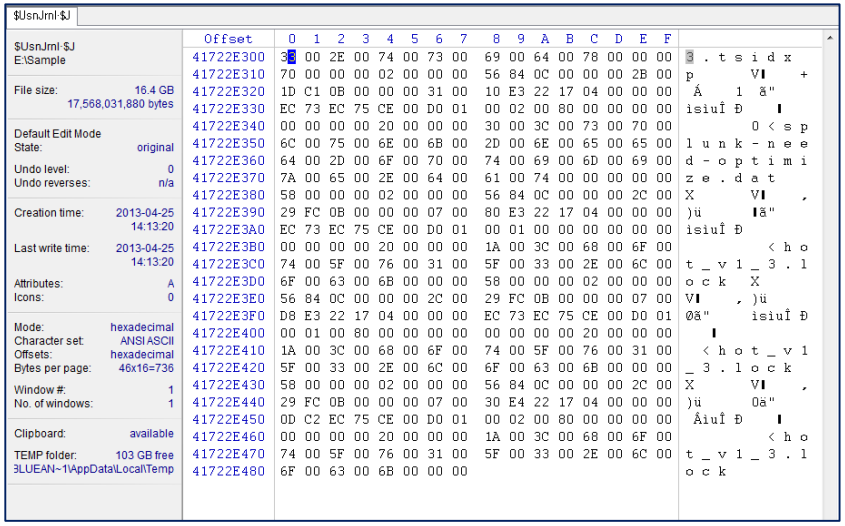
- Encase 에서 확인 시, Logical Size 이후에도 데이터 기록됨

	Name	Initialized Size	Logical Size	Physical Size
1	$ObjId	0	0	0
2	$ObjId:$O	393,216	393,216	393,216
3	$Quota	0	0	0
4	$Quota:$O	88	88	88
5	$Quota:$Q	208	208	208
6	$Reparse	0	0	0
7	$Reparse:$R	4,096	4,096	4,096
8	$RmMetadata	336	336	336
9	$UsnJrnl	0	0	0
10	$UsnJrnl:$J	17,568,031,880	17,568,031,880	17,568,301,056
11	$UsnJrnl:$Max	32	32	32



- Encase 기본 추출 옵션은 Logical File Only...





$UsnJrnl 수집

- **Encase**
  - "Entire Physical File" 옵션 선택 후, 추출

- **Winhex**
  - 기본적으로 Physical Size 로 추출됨

- **ExtractUsnJrnl (https://github.com/jschicht/ExtractUsnJrnl)**
  - Sparse 영역을 제외한 실제 데이터 영역만 추출

$J 속성의 로그 레코드 구조(http://msdn.microsoft.com/en-us/library/aa365722.aspx)

Offset	Size	Stored Information	Detail
0x00	4	Size of Record	레코드 크기
0x04	2	Major Version	2(현재 일반적으로 사용되는 Change Journal Software의 버전은 2.0)
0x06	2	Minor Version	0(현재 일반적으로 사용되는 Change Journal Software의 버전은 2.0)
0x08	8	MFT Reference Number	현재 변경 이벤트가 적용되는 파일 혹은 디렉터리의 MFT Reference Number
0x10	8	Parent MFT Reference Number	현재 변경 이벤트가 적용되는 파일 혹은 디렉터리의 부모 디렉터리의 MFT Reference Number $MFT 정보와 조합하여 전체 경로 획득 가능
0x18	8	USN	Update Sequence Number
0x20	8	TimeStamp(FILETIME)	이벤트가 발생한 시간(UTC +0)
0x28	4	Reason Flag	변경 이벤트 정보 플래그
0x2C	4	Source Information	변경 이벤트를 발생시킨 주체에 대한 정보
0x30	4	Security ID	보안 ID
0x34	4	File Attributes	변경 이벤트의 대상이 되는 객체에 대한 정보 일반적으로 대상이 파일인지 디렉터리인지 구분
0x38	2	Size of Filename	객체 이름 정보의 크기
0x3A	2	Offset to Filename	객체 이름 정보의 레코드 내 위치
0x3C	N	Filename	현재 변경 이벤트가 적용되는 객체(파일 혹은 디렉터리)의 이름

- MFT Reference Number 대신 Parent MFT Reference Number 를 사용하는 이유
  - ✓ MFT Reference Number 를 사용할 경우, 해당 파일이 삭제되었을 때 전체 경로를 못 얻을 수도 있기 때문

Reason Flag 정보(http://msdn.microsoft.com/en-us/library/aa365722.aspx)

Flag	Description
0x01	기본 $Data 속성에 데이터가 Overwrite 됨
0x02	기본 $Data 속성에 데이터가 추가됨
0x04	기본 $Data 속성에 데이터가 줄어듬
0x10	이름 있는 $Data 속성에 데이터가 Overwrite 됨
0x20	이름 있는 $Data 속성에 데이터가 추가됨
0x40	이름 있는 $Data 속성에 데이터가 줄어듬
0x100	파일이나 디렉터리가 생성됨
0x200	파일이나 디렉터리가 삭제됨
0x400	파일의 확장된 속성이 변경됨
0x800	접근 권한이 변경됨
0x1000	객체명 변경시, 변경 전 이름
0x2000	객체명 변경시, 변경 후 이름
0x4000	인덱스 상태가 변경됨
0x8000	파일이나 디렉터리의 속성이 변경됨
0x10000	하드 링크가 생성되었거나 삭제됨
0x20000	압축 상태가 변경됨(압축됨 or 압축이 풀림)
0x40000	암호화 상태가 변경됨(암호화됨 or 복호화됨)
0x80000	객체 ID가 변경됨
0x100000	Reparse 지점값이 변경됨
0x200000	이름 있는 $Data 속성의 생성 or 삭제 or 변경됨
0x80000000	파일 또는 디렉터리가 닫힘

Source Information 정보(http://msdn.microsoft.com/en-us/library/aa365722.aspx)

Flag	Description
0x00	사용자가 발생시킨 이벤트
0x01	운영체제에 의해 발생한 이벤트
0x02	The operation adds a private data stream to a file or directory.
0x04	The operation creates or updates the contents of a replicated file.

File Attribute 정보(http://msdn.microsoft.com/en-us/library/gg258117.aspx)

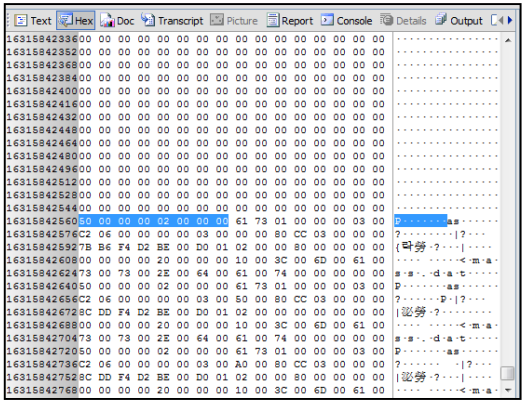
Value	Description
0x01	읽기 전용 속성
0x02	숨김 속성
0x04	시스템 파일
0x10	디렉터리
0x20	Archive 파일
0x40	디바이스 파일
0x80	일반 파일
0x100	임시 파일
0x200	Sparse 파일
0x400	Reparse 속성을 가지고 있거나 심볼릭 링크 파일
0x800	압축됨
0x1000	This attribute indicates that the file data is physically moved to offline storage.
0x2000	인덱싱 안됨
0x4000	암호화됨
0x8000	The directory or user data stream is configured with integrity (only supported on ReFS volumes).
0x10000	가상 파일
0x20000	The user data stream not to be read by the background data integrity scanner (AKA scrubber).

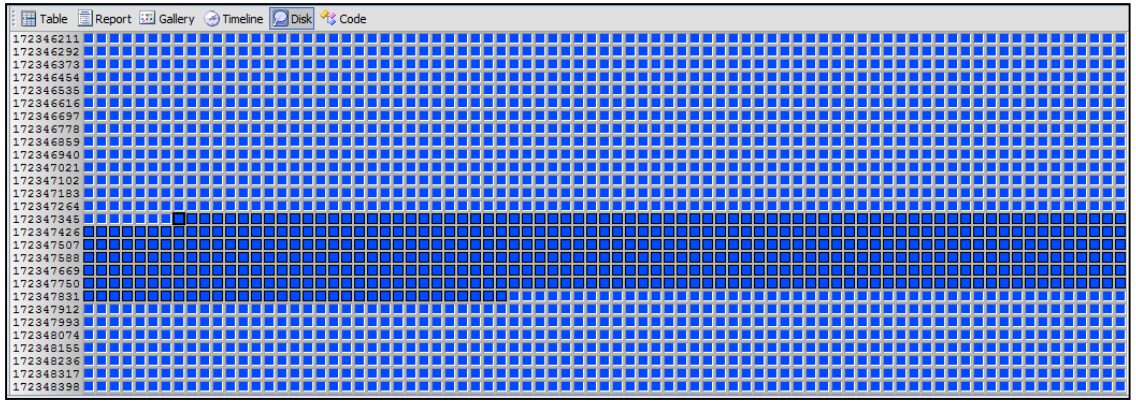
$UsnJrnl Record Carving

$UsnJrnl Record Carving

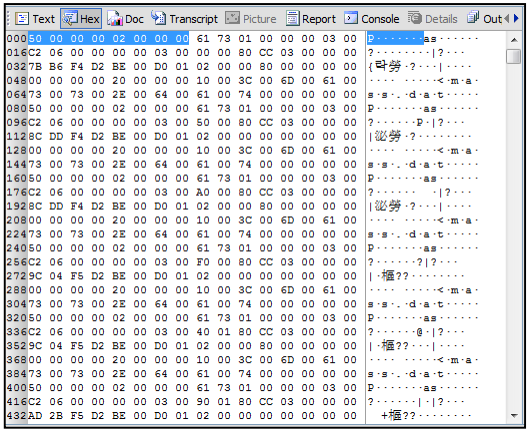
비할당 영역에 남아 있는 UsnJrnl 레코드

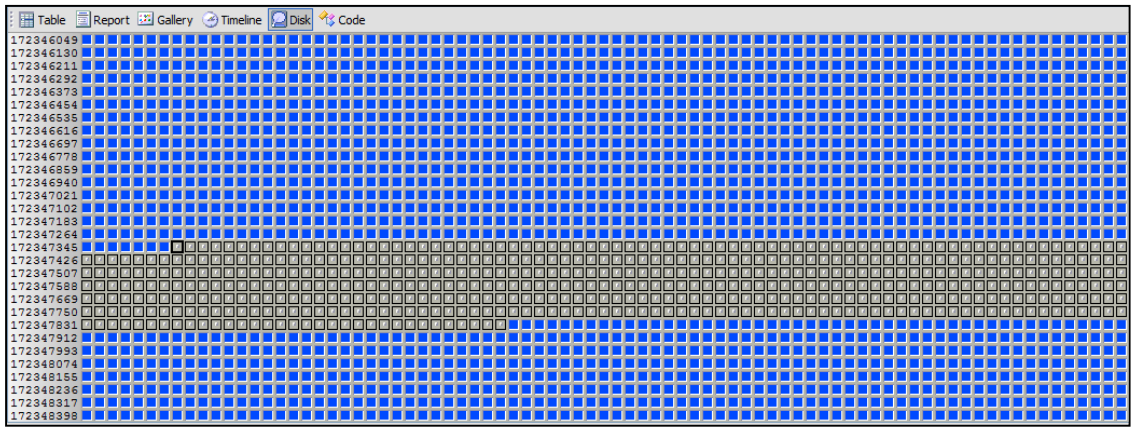
- **$UsnJrnl:$J 내 첫 레코드의 디스크 내 위치 : 172347352 sector**





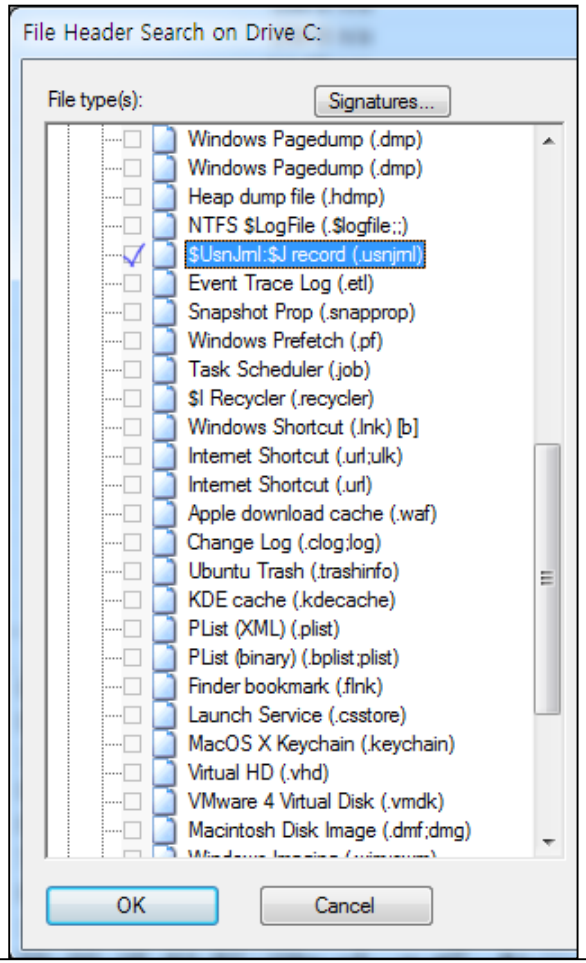
- **시간이 지난 후의 해당 디스크 영역 상태 ➔ 데이터는 그대로 있고 비할당 영역으로 변경됨**

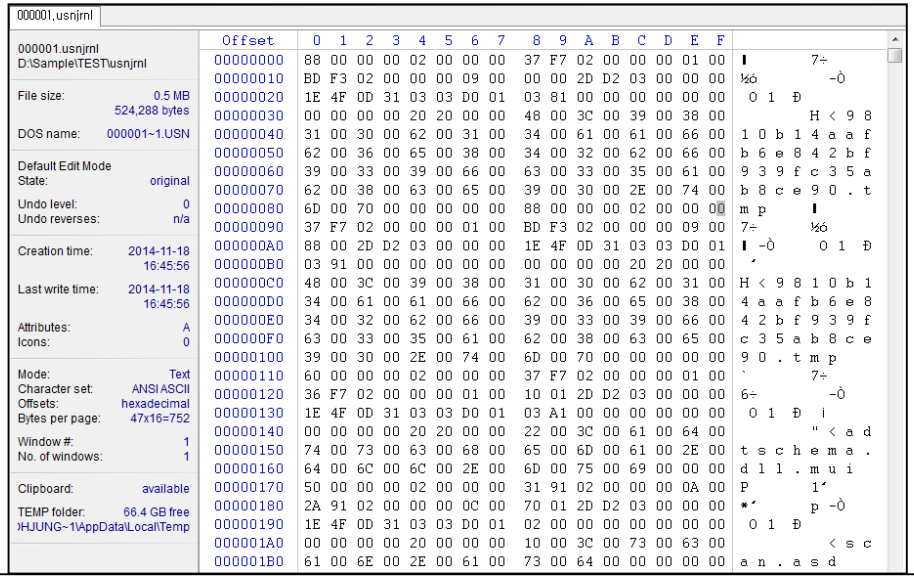




기존 도구 1

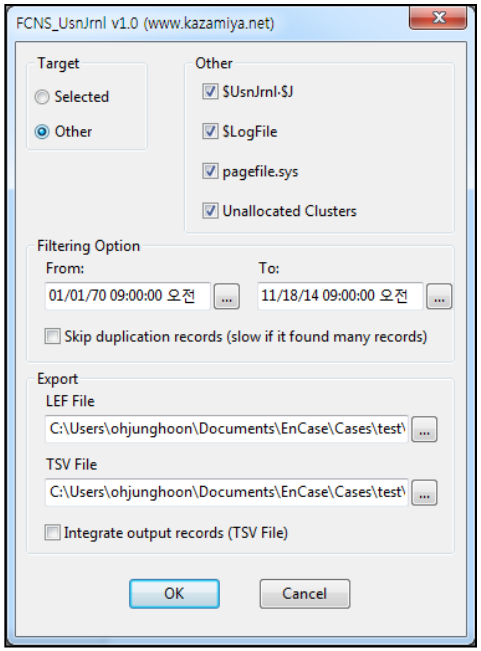
- **X-way forensics 의 USN Record 카빙 기능(Tool➔Disk Tool➔File Recovery by Type)**
  - 레코드 카핑 후, 여러 개의 레코드를 묶어서 파일 단위로 저장함





기존 도구 2

- **FCNS_UsnJrnl EnScript**
  - Encase 7 용 EnPack
  - 전체 경로 정보가 없음
  - L01, CSV 출력
    - ✓ L01 은 읽다가 시스템 다운(EnCase7;;)
    - ✓ CSV 파일 분할 기능이 없음(한 번에 다 못 읽음;;)



이름	수정한 날짜	유형	크기
FCNS_UsnJrnl_Data.L01	2014-11-18 오후...	EnCase Logical Evidence File	2,096,995KB
FCNS_UsnJrnl_Data.L02	2014-11-18 오후...	L02 파일	19,570,272KB
FCNS_UsnJrnl_Records.csv	2014-11-18 오후...	Microsoft Excel 쉼표로 구분된 값 파일	1,177,528KB

ItemPath	PS	SO	TimeStamp	FileName	FileID	ParentID	Reason	Reason(String)
C₩Unallocated Clusters	0	0		1				
pnidevdolocsemfrmotblopoahshgupgua	6881385	655471	000a0030	ADS				
C₩Unallocated Clusters	3459424	400	11/17/14 03:15:05 오후	aosmon.log	162023	25466	80000002	DATA
C₩Unallocated Clusters	3459426	176	11/17/14 03:15:05 오후	PaLogU.log	141286	141610	80000002	DATA
C₩Unallocated Clusters	3459426	448	11/17/14 03:15:05 오후	agent.dbf-journal	39515	141611	103	CREATE
C₩Unallocated Clusters	3459427	32	11/17/14 03:15:05 오후	agent.dbf	124844	141611	1	DATA
C₩Unallocated Clusters	3459427	480	11/17/14 03:15:05 오후	SendLog.apc	39515	141611	80000102	CREATE
C₩Unallocated Clusters	3459428	136	11/17/14 03:15:05 오후	PaSvc.log	192015	141610	3	DATA
C₩Unallocated Clusters	3459428	216	11/17/14 03:15:06 오후	SendLog.apc	39515	141611	80000200	DELETE
C₩Unallocated Clusters	3459429	32	11/17/14 03:15:06 오후	PaLogU.log	141286	141610	80000002	DATA
C₩Unallocated Clusters	3459429	112	11/17/14 03:15:06 오후	agent.dbf	124844	141611	80000001	DATA
C₩Unallocated Clusters	3459430	400	11/17/14 03:15:06 오후	PaLogU.log	141286	141610	80000002	DATA
C₩Unallocated Clusters	3459431	160	11/17/14 03:15:06 오후	agent.dbf-journal	39515	141611	103	CREATE

레코드 카빙

Offset	Size	Stored Information	Detail
0x00	4	Size of Record	레코드 크기
0x04	2	Major Version	2(현재 일반적으로 사용되는 Change Journal Software의 버전은 2.0)
0x06	2	Minor Version	0(현재 일반적으로 사용되는 Change Journal Software의 버전은 2.0)
0x08	8	MFT Reference Number	현재 변경 이벤트가 적용되는 파일 혹은 디렉터리의 MFT Reference Number
0x10	8	Parent MFT Reference Number	현재 변경 이벤트가 적용되는 파일 혹은 디렉터리의 부모 디렉터리의 MFT Reference Number $MFT 정보와 조합하여 전체 경로 획득 가능
0x18	8	USN	Update Sequence Number
0x20	8	TimeStamp(FILETIME)	이벤트가 발생한 시간(UTC +0)
0x28	4	Reason Flag	변경 이벤트 정보 플래그
0x2C	4	Source Information	변경 이벤트를 발생시킨 주체에 대한 정보
0x30	4	Security ID	보안 ID
0x34	4	File Attributes	변경 이벤트의 대상이 되는 객체에 대한 정보 일반적으로 대상이 파일인지 디렉터리인지 구분
0x38	2	Size of Filename	객체 이름 정보의 크기
0x3A	2	Offset to Filename	객체 이름 정보의 레코드 내 위치
0x3C	N	Filename	현재 변경 이벤트가 적용되는 객체(파일 혹은 디렉터리)의 이름

- 시그니처 : ₩x??₩x??₩x00₩x00₩x02₩x00₩x00₩x00
- 서브 체크 포인트 : USN, TimeStamp, Source Information, Size/Offset of Filename

레코드 카빙 결과

시스템	비할당 영역 크기	카빙 레코드 수(중복제거)	레코드 시간 범위
A(Win7 64bit)	72G(HDD)	32,379,635	2014-02-10 ~ 2014-11-03
B(Win7 64bit)	120G(HDD)	36,650,278	2014-01-28 ~ 2014-11-10
C(Win7 64bit)	269G(HDD)	24,907,010	2014-01-28 ~ 2014-11-13
D(Win7 64bit)	120G(HDD)	22,310,563	2013-10-27 ~ 2014-12-23

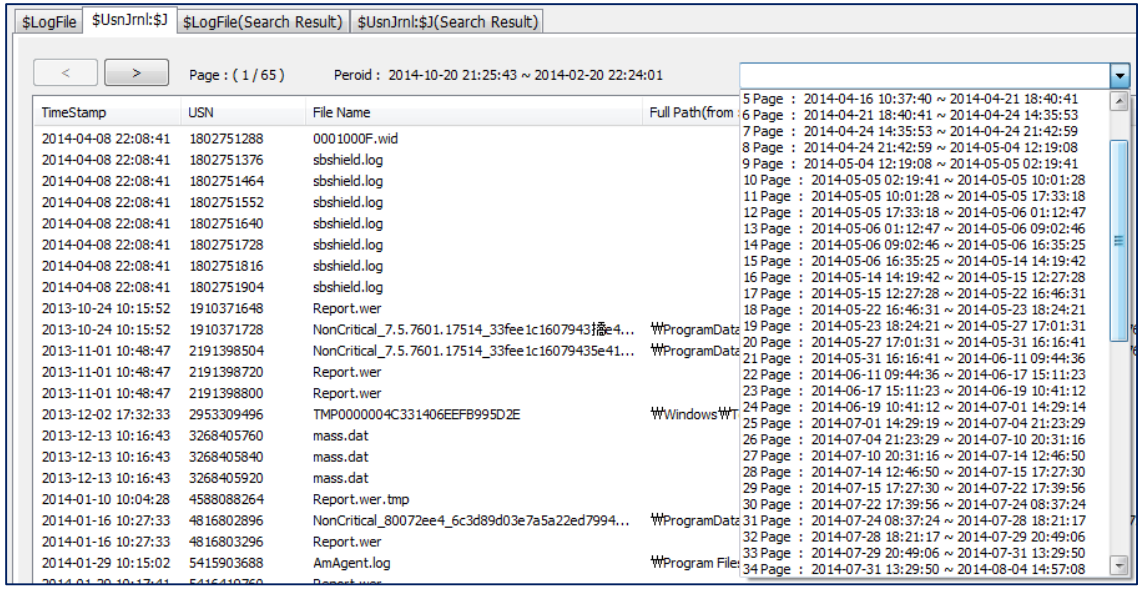
- **평균 3천만 개 내외의 레코드가 추출됨**
  - 일반적으로 추출한 $UsnJrnl:$J 파일에는 30 만 개 정도의 레코드가 존재함
  - 10~11 개월 이전의 레코드까지 발견됨

- **포멧 이전의 레코드들도 존재**
- **다른 매체(ex : USB) 의 레코드 흔적도 존재**
  - $LogFile 의 Page 내의 작업 흔적에 남아 있음
  - 메모리 내 데이터 혹은 임시 파일의 흔적으로 추정...

NTFS Log Tracker v1.4

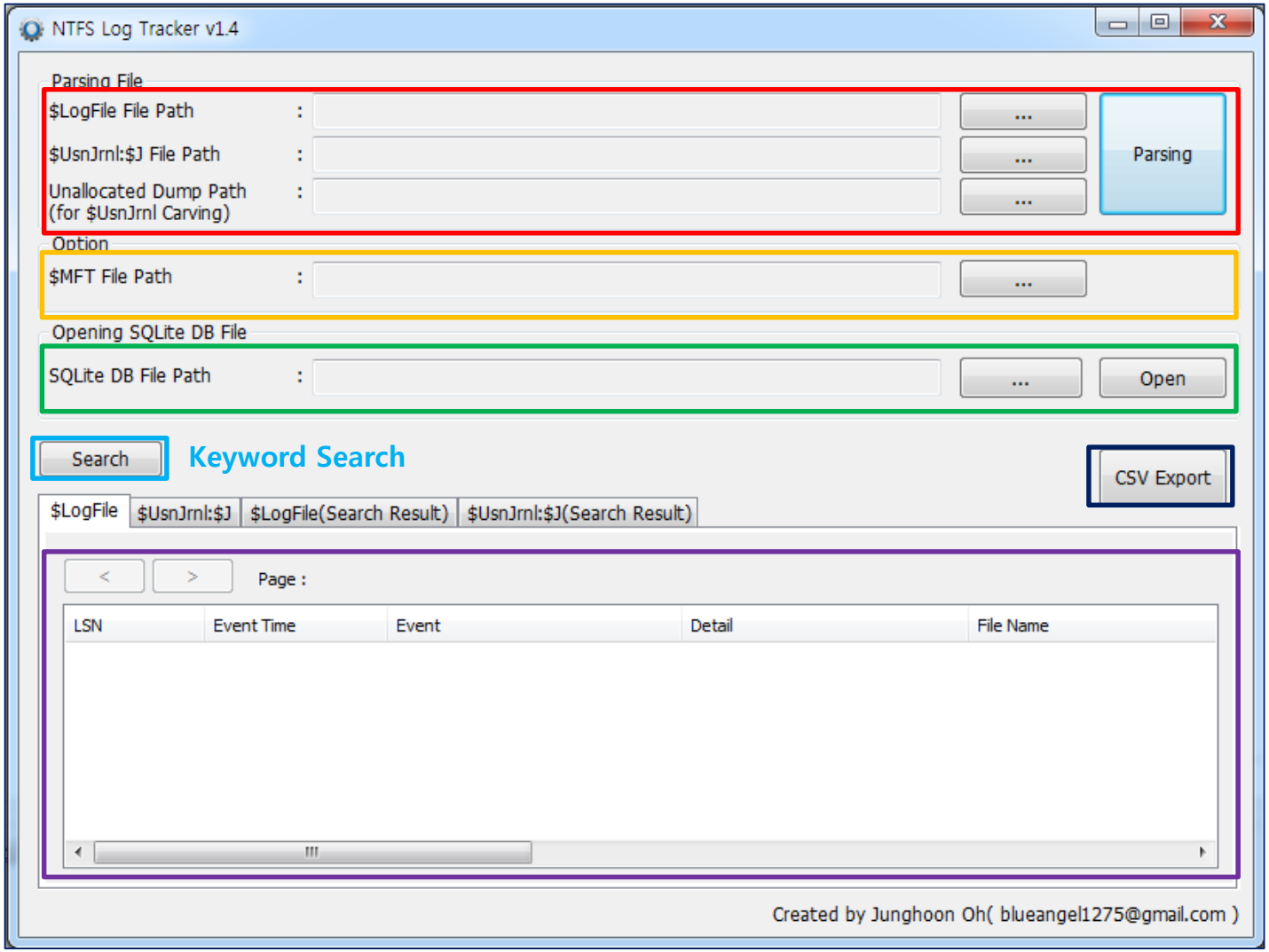
업데이트 내역(https://sites.google.com/site/forensicnote/ntfs-log-tracker)

1. **비할당영역 덤프 파일을 대상으로 $UsnJrnl 레코드 카빙**
   - 대량의 데이터를 페이지 단위로 출력(한 페이지당 500,000 레코드 출력)
   - $MFT 정보를 바탕으로 전체 경로(Full Path) 정보 추가
   - 페이지 인덱싱 작업 수행(3 페이지 이상일 경우)
     - ✓ USN 순으로 정렬 후, 각 페이지 별 첫 레코드와 마지막 레코드의 시간 정보 기록



2. **ExtractUsnJrnl 를 통해 수집된 파일도 파싱 지원**
3. **키워드 검색 인터페이스 변경**
4. **한글 키워드 검색 지원**
5. **Tab 버그 수정**

도구 유저 인터페이스



Source File for Parsing

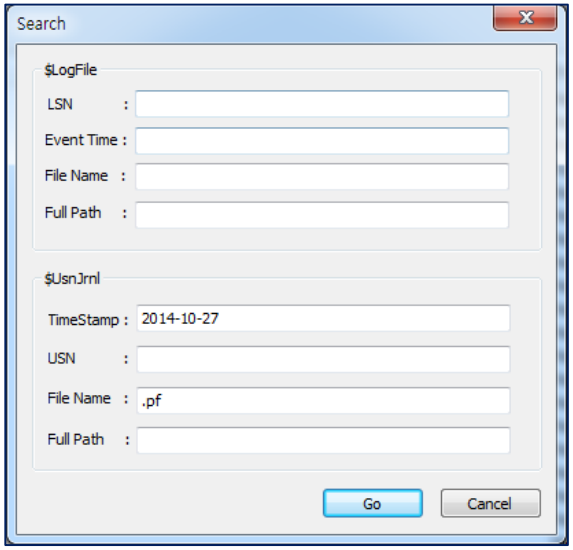
$MFT for Full Path Construction

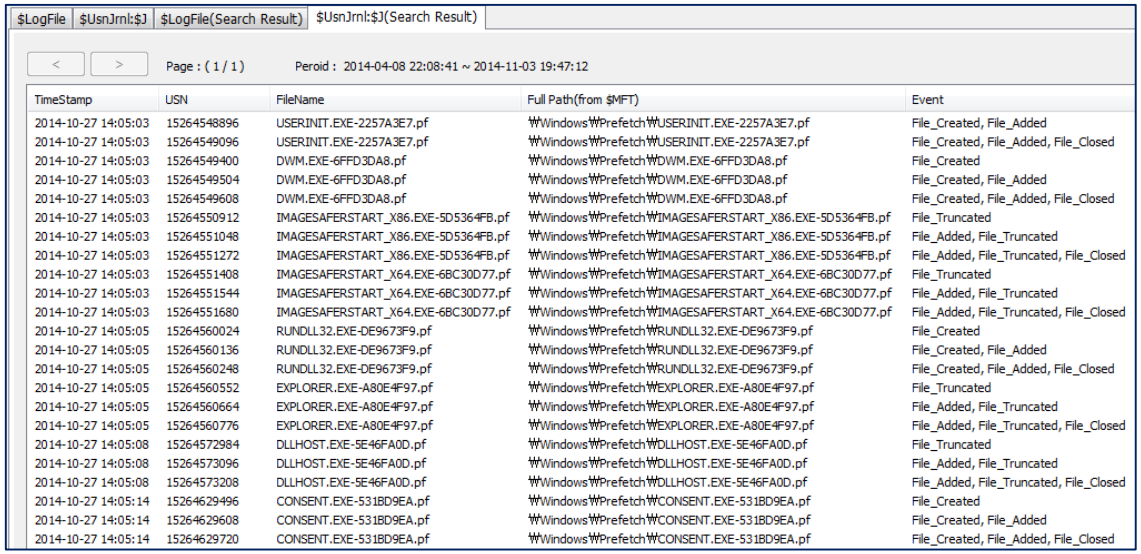
Opening DB file created by this tool

Exporting CSV format

Parsed Data Output

키워드 검색





- 키워드 검색 대상 필드
  - ✓ $LogFile : LSN, Event Time, File Name, Full Path
  - ✓ $UsnJrnl : TimeStamp, USN, File Name, Full Path
- SQL 의 LIKE 연산자 사용
- 복수 키워드 입력 시, AND 연산으로 수행됨

Conclusion

Conclusion

- **$UsnJrnl 을 통한 파일 시스템 히스토리 추적**
  - 파일/디렉터리의 생성/삭제/수정 이벤트 추적
  - 삭제된 파일의 히스토리 추적
  - 프리패치 파일(.pf), 링크파일(.lnk) 흔적을 통한 프로그램 실행 및 문서 열람 정보 확인

- **$UsnJrnl:$J 수집**
  - Encase or Winhex
  - ExtractUsnJrnl

- **비할당 영역에서의 $UsnJrnl 레코드 카빙**
  - 대량의 $UsnJrnl 레코드가 비할당영역에 남아 있음
  - 레코드 카빙을 통한 오래된 파일 시스템 히스토리 추적

Question and Answer