10장. 디지털 포렌식(2)

박종혁

서울과학기술대학교 컴퓨터공학과

jhpark1@seoultech.ac.kr

개요

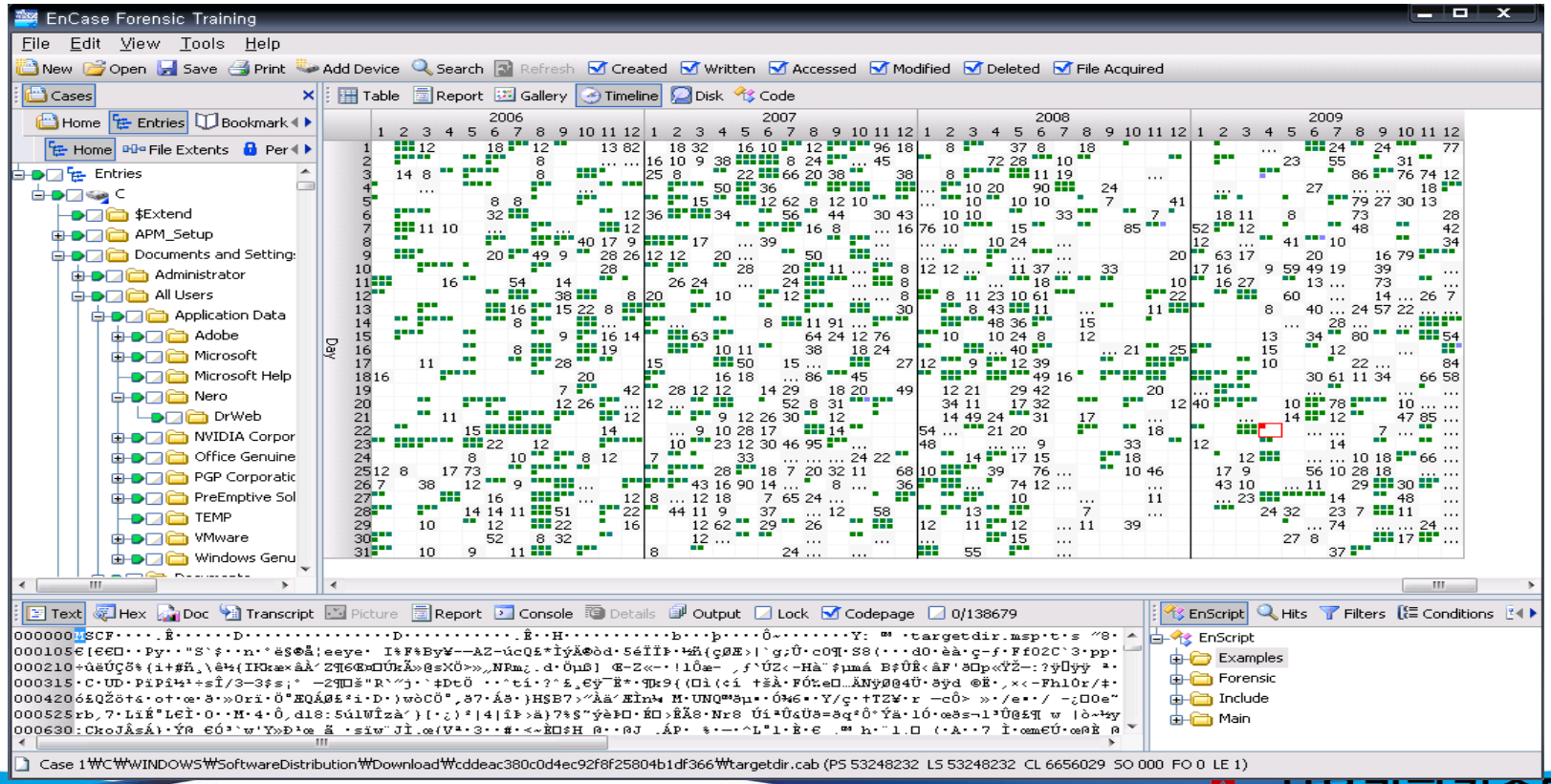
- **학습 목표**
  - 수집된 디지털 데이터를 분석하여 사건의 실마리 또는 증거를 찾기 위한 다양한 기술들을 살펴본다.
  - 디지털 증거 분석을 데이터 뷰잉, 검색, 통계 분석, 타임라인 분석 기술 등 다양한 기술에 대해 살펴본다.
- **학습 내용**
  - 다양한 증거 분석 기술
  - 증거 분석 도구를 활용한 증거 분석
  - 안티 포렌식 대응 기술

목 차

1. 타임라인 분석

타임라인 분석

- **타임라인 분석**
  - 디지털 데이터의 시간 정보는 범죄 사실을 규명하기 위해 매우 중요한 정보
  - 파일 시스템 상에 저장되는 파일의 시간 정보, 파일 내부의 메타데이터에 저장되는 시간 정보 등 다양한 곳에 저장되어 있는 시간 정보를 이용,타임라인(Timeline)을 구성함으로써 시스템 사용자의 행위를 추적할 수 있음



2. 로그 분석

로그 분석

- **로그(LOG)란?**
  - 시스템에 접속한 사용자의 행위 및 시스템의 상태를 주기적으로 저장해 놓은 기록
  - 로그를 이용하여 외부 침입의 흔적과 사용자가 어떠한 명령어를 사용했는지, 그리고 시스템이 처리한 업무와 에러 등의 정보 등을 파악
  - 서버 시스템의 침해사고조사와 같은 경우 가장 기본적으로 행해지는 분석 중의 하나

- **로그의 종류**
  - Unix 시스템 계열 로그, Windows 계열 로그, 웹(Web) 로그 등
  - 시스템의 종류에 따라 특별한 설정 없이 기본적으로 생성되는 로그가 있는 반면, 사용자의 설정이 있어야만 생성되는 로그도 존재
  - 조사자는 각 시스템의 기본 로그와 그렇지 않은 로그의 분석 방법을 숙지해야 함

로그분석 - Unix 시스템 로그 분석

- 시스템 별 로그 디렉터리

Unix 시스템	디렉터리
HP-UX	/usr/adm
Solaris, AIX	/var/adm
Linux, BSD	/var/log

- 로그 파일의 종류 및 기본적인 기능

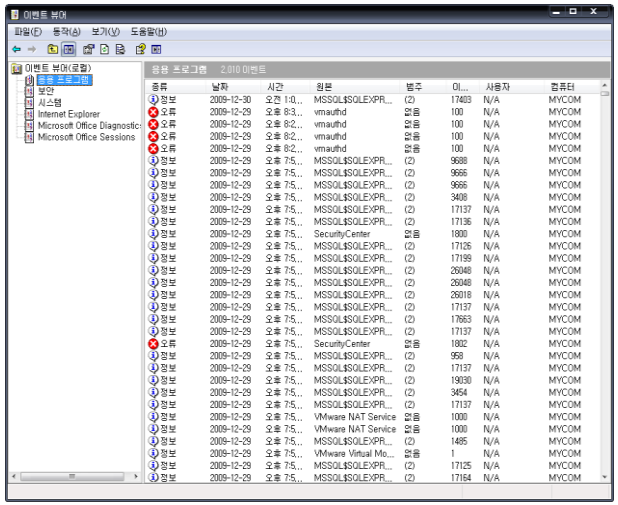
파 일 명	기 능
acct 또는 pacct	사용자별로 실행되는 모든 명령어를 기록
aculog	다이얼-아웃 모뎀 관련 기록(자동 호출 장치)
lastlog	각 사용자의 가장 최근 로그인 시간을 기록
loginlog	실패한 로그인 시도를 기록
messages	부트 메시지 등 시스템의 콘솔에서 출력된 결과를 기록하고 syslog에 의하여 생성된 메시지도 기록
sulog	su 명령 사용 내역 기록
utmp	현재 로그인한 각 사용자의 기록
utmpx	utmp 기능을 확장(extended utmp), 원격 호스트 관련 정보 등 자료 구조 확장
wtmp	사용자의 로그인, 로그아웃 시간과 시스템의 종료 시간, 시스템 시작 시간 등을 기록
wtmpx	wtmp 기능 확장 (extended wtmp)
vold.log	플로피 디스크나 CD-ROM과 같은 외부 매체의 사용에서 발생하는 에러를 기록
xferlog	FTP 접근 기록

로그분석 - Windows 시스템 로그 분석

- **이벤트 로그**
  - Windows는 기본적으로 이벤트(event) 로그를 시스템 운영 전반에 걸쳐서 저장
  - 조사자는 이벤트 로그의 분석을 통해 해당 시스템의 전반적인 동작을 알 수 있으며, 증거 자료를 획득할 수도 있음

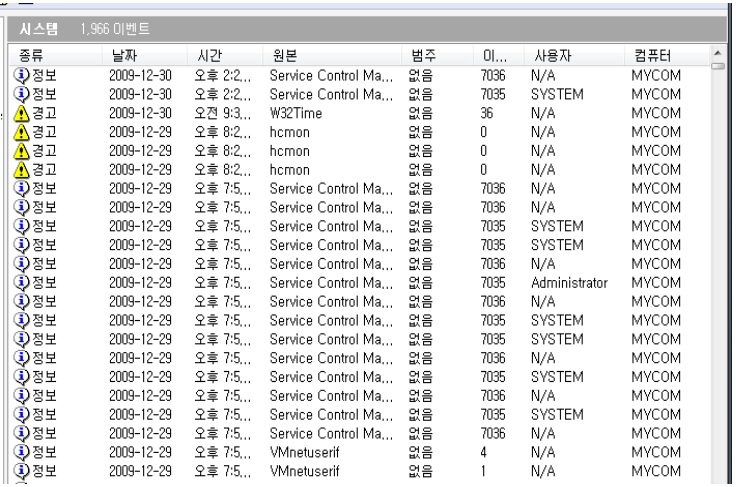
- **이벤트 로그의 종류**
  - 응용프로그램 로그
    - 응용프로그램이나 기타 프로그램의 동작에 대한 이벤트가 저장되며, 기록되는 이벤트는 소프트웨어 개발자에 의해 결정
  - 보안 로그
    - 유효하거나 유효하지 않은 로그온 시도 및 파일 생성, 열람, 삭제 등에 관련된 이벤트를 기록
  - 시스템 로그
    - Windows 시스템 구성요소가 기록하는 이벤트로 시스템 부팅 시 드라이버가 로드되지 않는 경우와 같이 구성요소의 오류를 기록

로그분석 - Windows 시스템 로그 분석



이벤트 헤더 정보

정 보	의 미
날 짜	이벤트가 발생한 날짜
시 간	이벤트가 발생한 시간
사 용 자	이벤트를 발생시킨 사용자의 이름
컴 퓨 터	이벤트가 발생한 컴퓨터의 이름
원 본	이벤트를 기록한 소프트웨어 (이벤트가 일어난 프로세스)
이벤트ID	해당 원본의 특정 이벤트 유형을 식별하는 번호
범 주	이벤트의 원본에 의한 이벤트 분류로 주로 보안 로그에서 사용됨
종 류	이벤트 심각도의 분류로 오류, 정보, 경고, 성공, 감사, 실패 감사로 분류



이벤트 정보의 종류

이벤트 유형	설 명
오 류	데이터 손실이나 기능 상실 같은 중대한 문제로 시스템을 시작하는 동안 서비스가 로드되지 못했을 경우와 같은 이벤트 기록
경 고	시스템에 문제가 발생할 수 있는 문제를 미리 알려 주는 이벤트로 디스크 공간이 부족할 때와 같은 이벤트 기록
정 보	응용 프로그램, 드라이버 또는 서비스가 성공적으로 수행되었음을 설명하는 이벤트
성공감사	사용자가 시스템에 성공적으로 로그온 했을 경우와 같이 보안 이벤트가 성공했음을 나타냄
실패감사	사용자가 시스템에 로그온 실패했을 경우와 같이 보안 이벤트가 실패했음을 나타냄

로그분석 - Web 로그 분석

- 웹 서비스를 제공하는 웹 서버의 종류에 따라 다른 형태로 저장
- CLF(Common Log Format)로 파일을 생성, 웹 서버의 종류와 설정에 따라 조금씩 차이

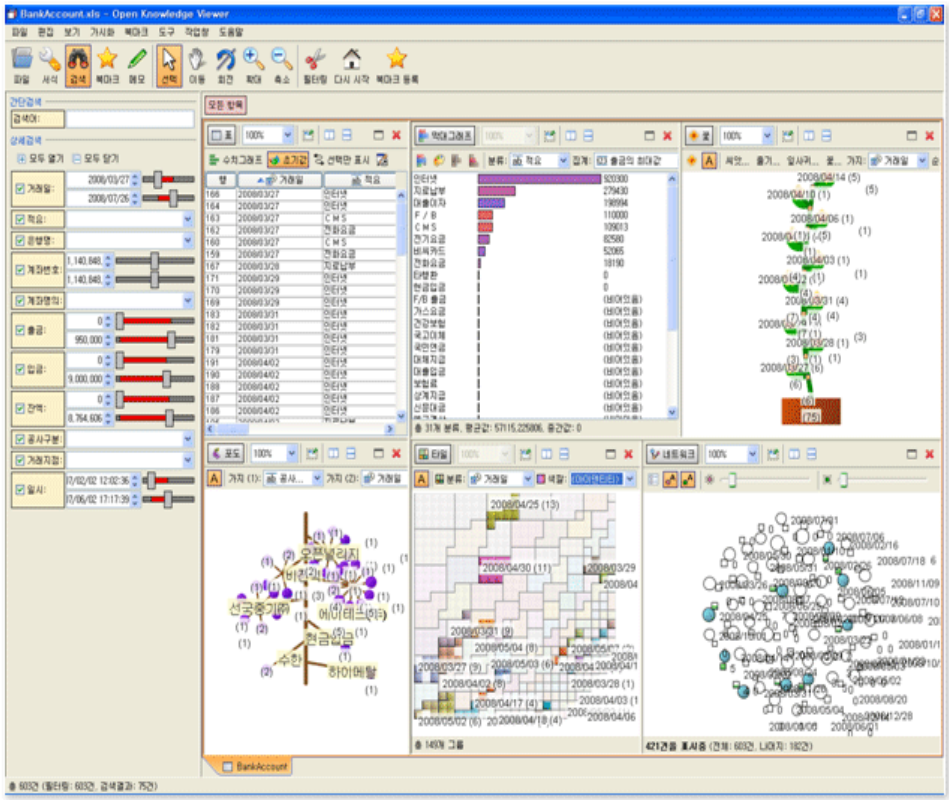
CLF 저장 정보 및 설명

로그 정보	설 명
Host	클라이언트의 호스트 이름이나 IP 주소
Authuser	인증이 필요한 경우 사용자 이름 기록
Date	접속한 시간과 날짜 기록
Request	클라이언트가 요청한 메시지
Status	요청한 것에 대한 서버의 처리사항 (상태 코드)
Bytes	전송된 Bytes의 크기 (헤더 제외)

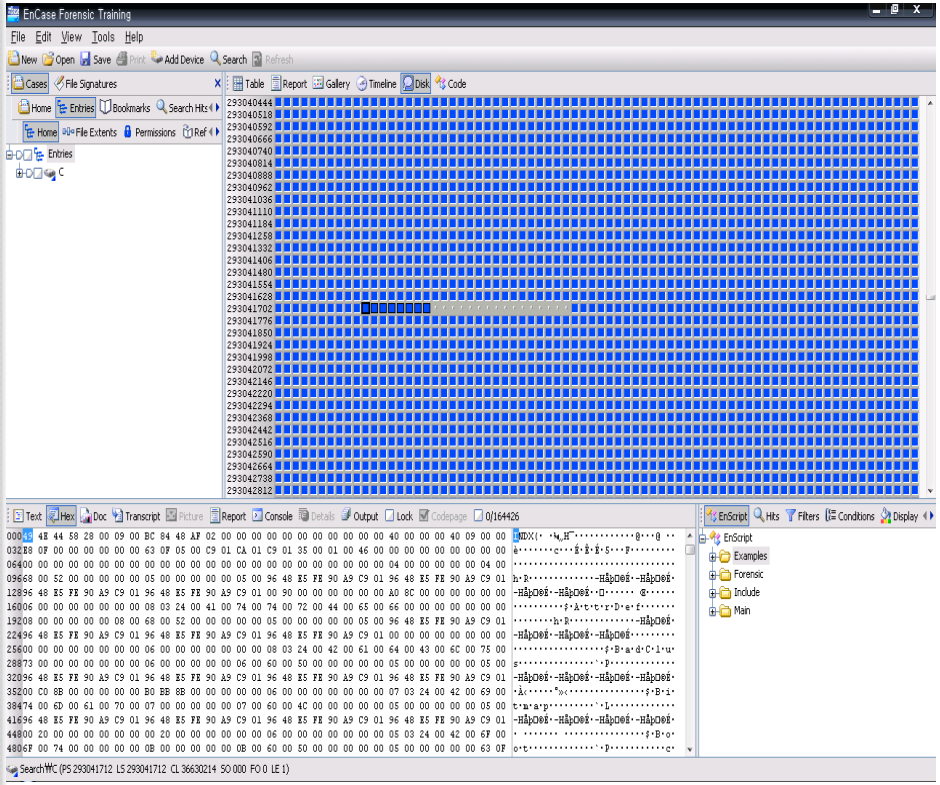
3. 시각화 기술

시각화 기술

- 보이지 않는 것을 일정한 형태로 나타내거나 가려져 있던 어떤 현상이나 실체가 눈에 띄게 드러나게 하는 것으로 추상적인 자료를 사람이 인지할 수 있는 형대로 만드는 과정



Open Knowledge Viewer



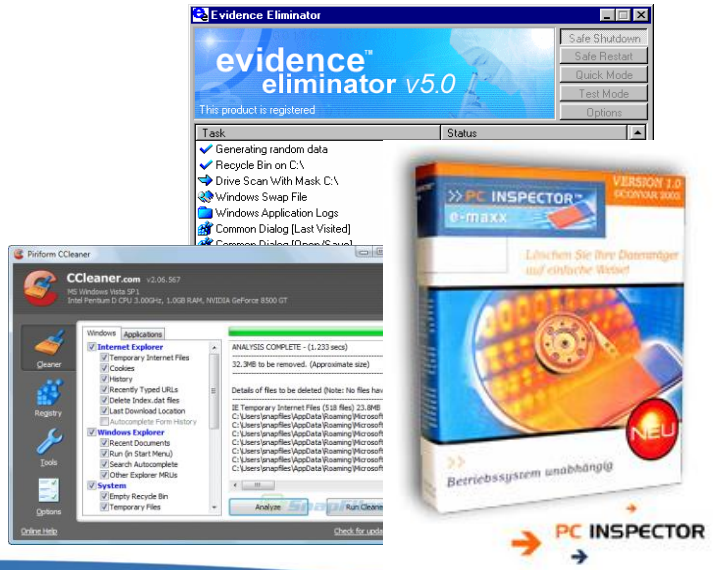
EnCase의 디스크 할당 상태 시각화 기능

4. 안티 포렌식 대응 기술

안티 포렌식 대응 기술

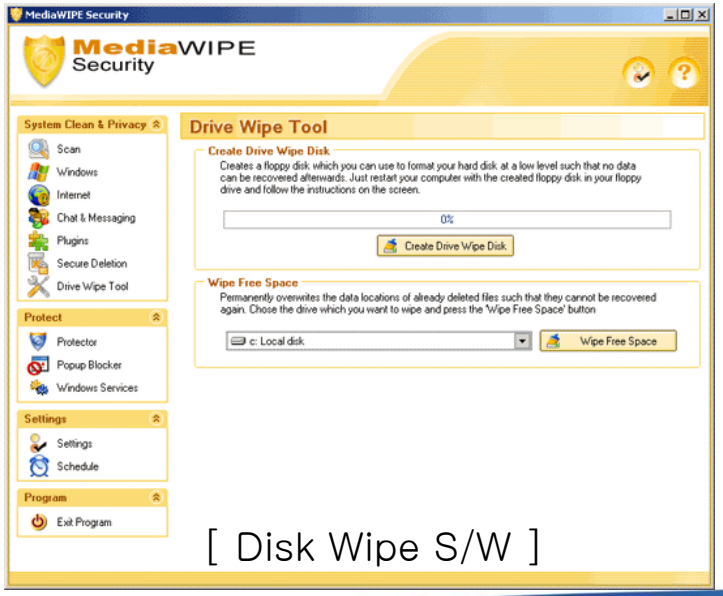
- **안티 포렌식 (Anti-Forensic)이란?**
  - 포렌식 기술에 대응하여 자신에게 불리하게 작용할 가능성이 있는 증거물을 차단하려는 일련의 활동
  - 과거에는 증거가 될 수 있는 자료들을 수동으로 처리하였지만, 최근에는 추적 및 증거물 획득을 원천적이고 자동화된 방법으로 막아주는 전문 제품들이 등장하고 있음

- **안티 포렌식 기법**
  - 주로 데이터 암호화 등을 통한 복구 기법 회피, 중요 증거 데이터의 증거 자동 삭제, 데이터 은닉 제품 등이 있음
  - 데이터 영구 삭제
    - Disk Wiping, Degausser
    - 증거 자동 삭제
  - 데이터 암호화
    - 압축파일, 문서파일 등의 암호화 등
  - 데이터 은닉
    - 스테가노그래피



안티 포렌식 대응 기술 - 데이터 영구 삭제

- **Disk Wipe**
  - 하드디스크의 기존 데이터를 완벽히 제거하고 모든 Sector의 내용을 0으로 만드는 과정
- **디가우저 (소자, Degausser)**
  - 하드디스크나 테이프에 강력한 자기장을 노출시켜 기록된 데이터를 파괴하고 복구가 불가능하도록 하는 장비



[Disk Wipe S/W]


소자용 전문 제품
(Degausser – BTE)

안티 포렌식 대응 기술 - 데이터 영구 삭제

- **데이터 복구 기법 회피 기술**
  - 디스크 덮어쓰기
    - 삭제된 파일의 데이터 중 물리적으로 디스크에 남아있는 부분을 덮어쓰고 삭제하는 과정을 반복하면 데이터 복구 기법을 회피할 수 있음
    - 美 국방성(DoD)에서는 기밀 자료를 삭제하기 위한 표준 (DoD5220, 22-M)을 다음과 같이 제시하고 있음

      1. 임의의 문자로 데이터를 덮어 씀
      2. 첫 번째 문자의 보수로 덮어씀
      3. 다시 임의의 문자로 데이터를 덮어씀
      4. 이 과정을 7회 반복

안티 포렌식 대응 기술 – 데이터 복구

삭제 파일 복구 기술의 필요성

- 비할당 영역에서 삭제 파일을 복구함으로써 심도 있는 컴퓨터 사용 흔적 조사가 가능
- 용의자 및 범죄자는 증거 인멸을 위해 저장 매체를 물리적으로 파괴, 훼손하거나 디지털 증거를 삭제할 가능성이 높으므로, 이를 원래의 상태로 복구하는 기술이 필요함

파일시스템의 영역

- **메타데이터 영역** : 파일의 이름, 만든이, 날짜, 크기 등을 저장
- **데이터 영역** : 파일의 실제 데이터 스트림 저장

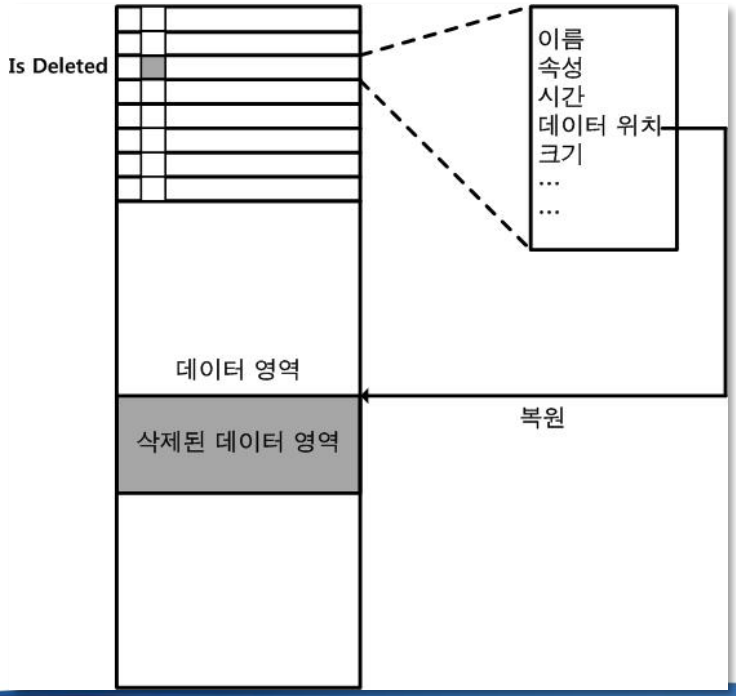
저장매체 복구 기술

- **물리적 복구** : 저장매체의 물리/전자적 복구. 물리적 또는 전자적 단/합선으로 훼손된 저장매체를 정상 상태로 복구하는 기술
- **논리적 복구** : 삭제/훼손된 파일 및 파일 시스템을 복구하는 기술

안티 포렌식 대응 기술 – 데이터 복구

파일시스템 메타데이터기반 복구

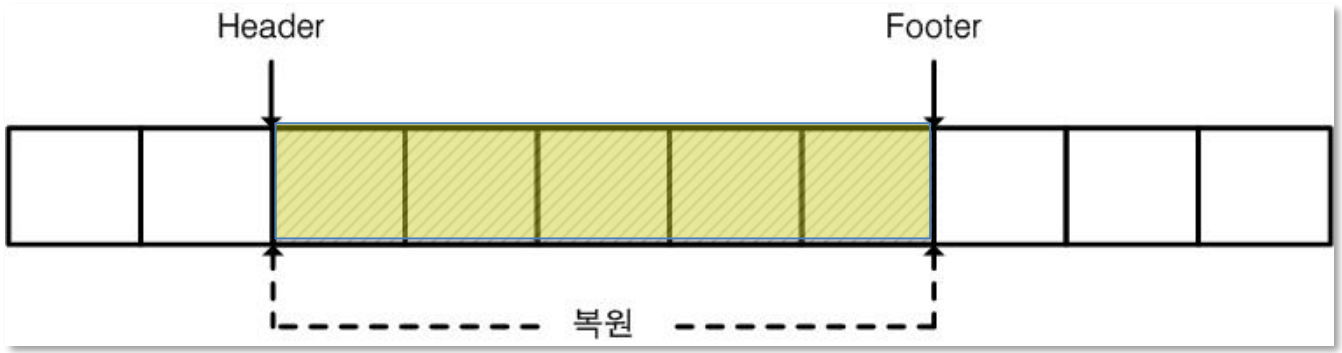
- 파일의 메타 데이터의 **'삭제 플래그'** 를 참조하여 복구
  - 파일 삭제 시 데이터 영역이나 메타데이터 영역을 덮어쓰지 않기 때문에 가능
  - 다른 파일로 덮이지 않았다면 복구 가능



안티 포렌식 대응 기술 – 데이터 복구

파일 시스템 정보를 얻을 수 없는 경우의 복구

- 파일 시스템에서 얻을 수 있는 정보 없이 **'파일 자체 정보'** 기반 복구
  - 즉, 파일의 고유한 특성이 있는 파일만 복구 가능
- 연속적으로 존재하는 파일에 대한 복구는 대부분 가능, 조각난 경우는 어려움
- 추출된 파일이 올바른 파일이라는 보장이 없음
- 많은 시간이 소요됨

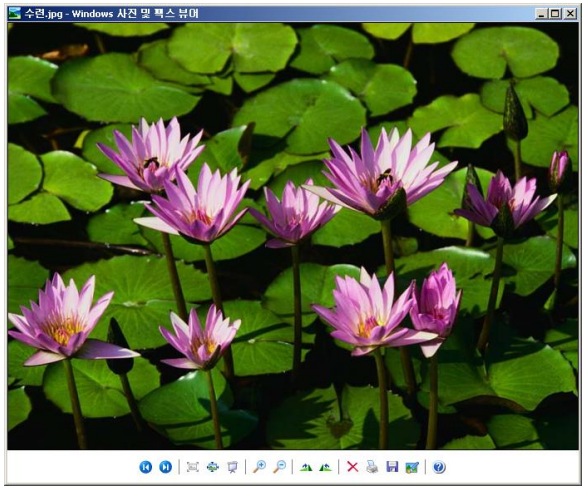


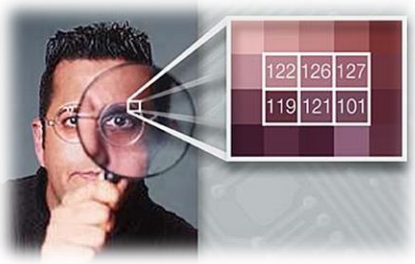
안티 포렌식 대응 기술 - 데이터 암호화

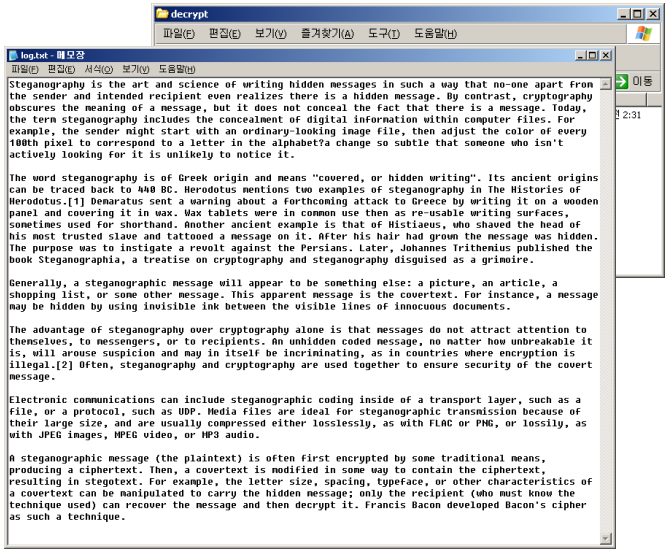
- **데이터 암호화**
  - Zip, Rar 등과 같은 압축파일에 암호화 기법을 적용하여, 증거확보를 어렵게 함
  - MS 오피스 및 한글 파일 등과 같은 문서를 암호화하여, 정보를 은폐하는데 활용되고 있음

안티 포렌식 대응 - 스테가노그라피

- **Steganography**
  - 메시지가 전송되고 있다는 사실 즉, 통신의 존재를 숨기는 기술로서 이미지 및 오디오 파일과 같은 다양한 디지털 매체를 통해 메시지를 은닉하여 전송하는 기술
    - 모르는 사람이 보면 평범한 사진에 불과하지만 약속된 수신자는 그 안에 메시지를 확인할 수 있는 기술

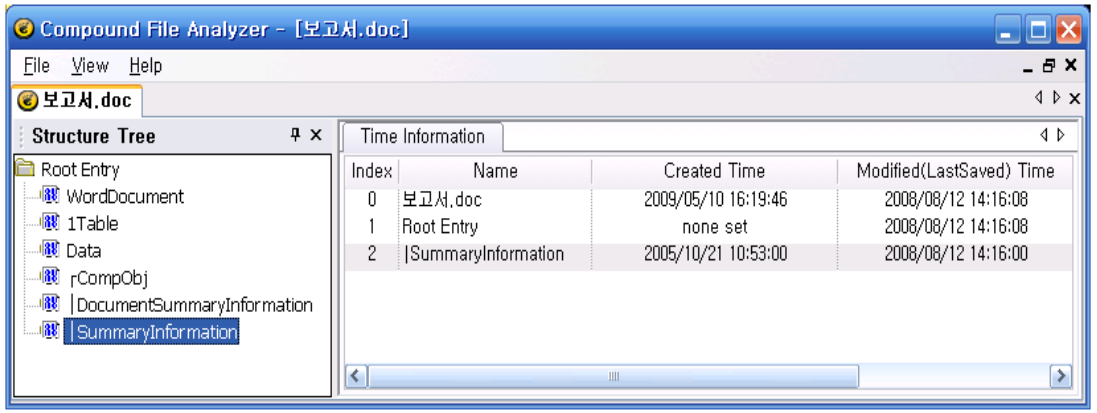






안티 포렌식 대응 기술 – 파일 내부의 시간 정보 분석

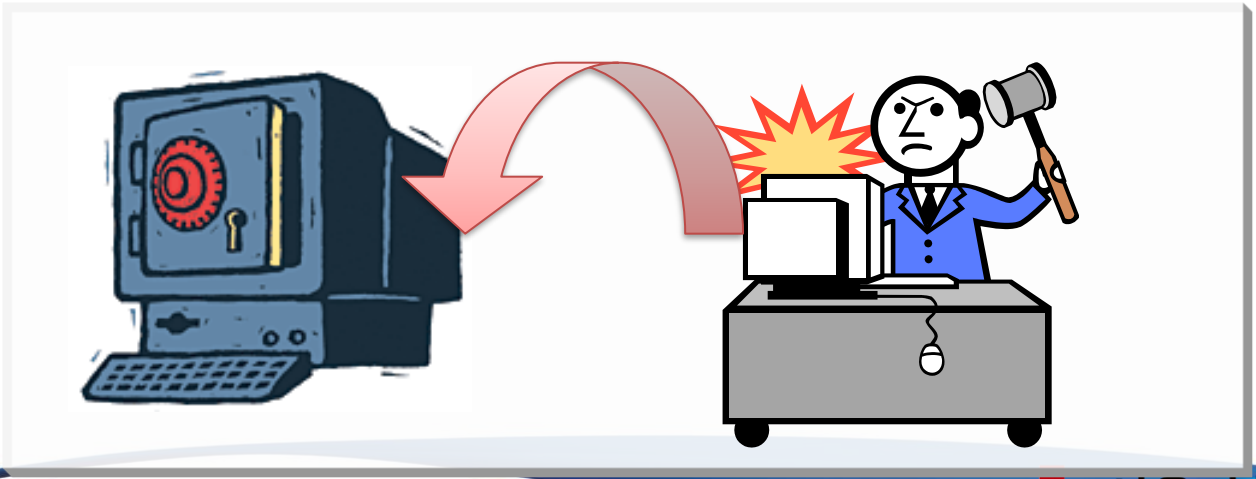
- 파일 시스템 상에 저장되는 시간 정보는 변조가 용이
- 응용 프로그램에 의해 파일 내부에 저장되는 시간 정보는 해당 파일의 저장 형식을 알지 못하면 변경하기 매우 어려움



문서 파일 내부 시간 정보 분석 도구 : CFA

안티 포렌식 기술 대응방안

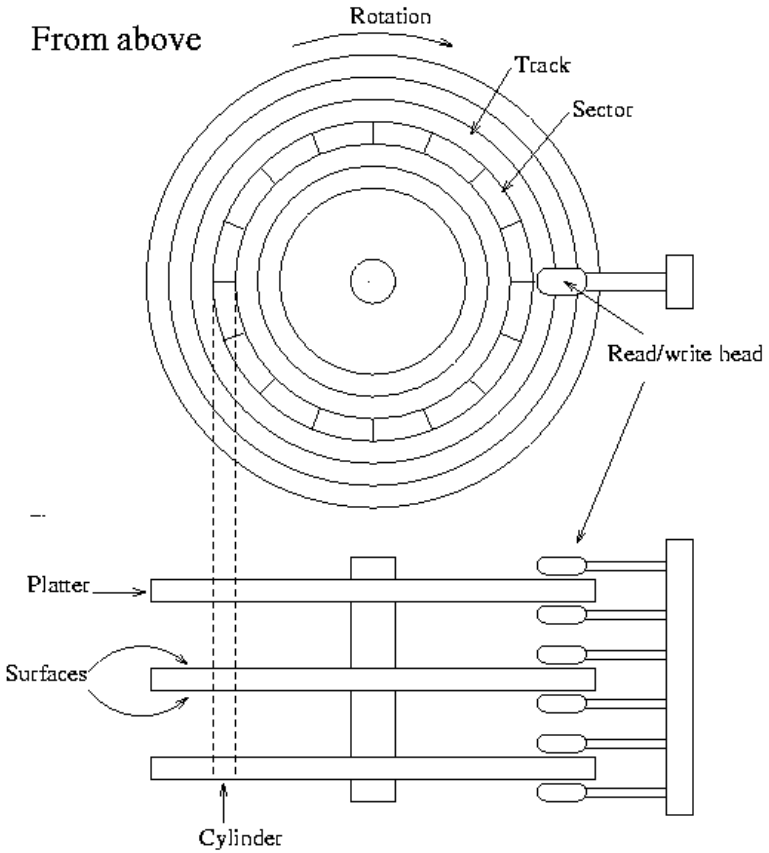
- **Anti-Forensic에 대한 대응**
  - 프라이버시 및 개인 정보 보호라는 긍정적인 측면도 있지만, 범죄자가 범행직후 증거를 없애는 용도로 사용하는 경우에는 컴퓨터 범죄 수사에 많은 어려움을 초래할 수 있음
  - Anti-Forensic 기술은 앞으로도 계속 발전되고 대중화 될 것이 예상되며, 이에 대응할 수 있는 기술과 정책적 기반이 필요함



5. 파일시스템의 이해

파일 시스템의 이해

- Hard disk의 구성

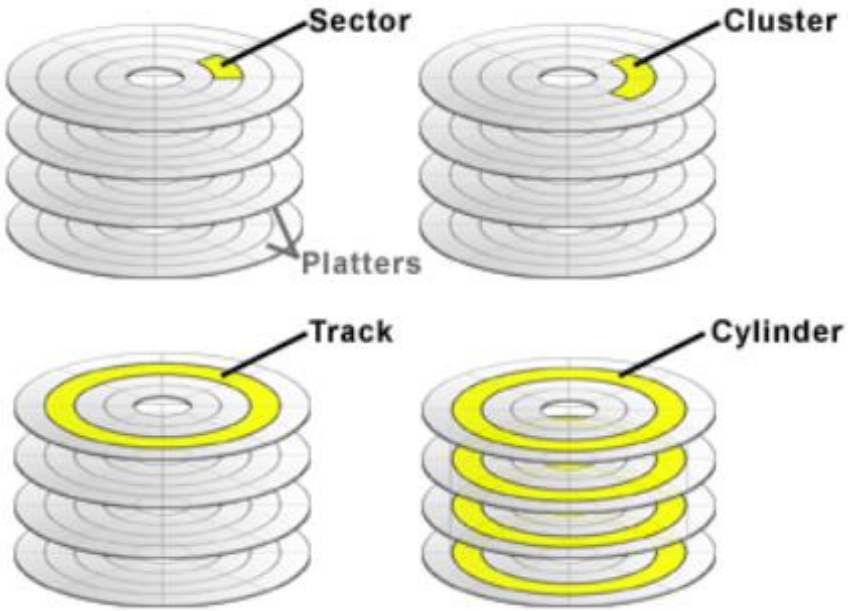


파일 시스템의 이해

- 플래터(platter): 비 자성체인 비금속 원판 표면에 자성체인 산화 금속막을 양면에 도장한 것
- 실린더 (Cyinder): 다수개의 플래터의 트랙을 수직적으로 관통하는 3차원 적인 스택
- 보통 섹터당 512KB 동심원을 동일한 각도로 나누어 데이터를 기록

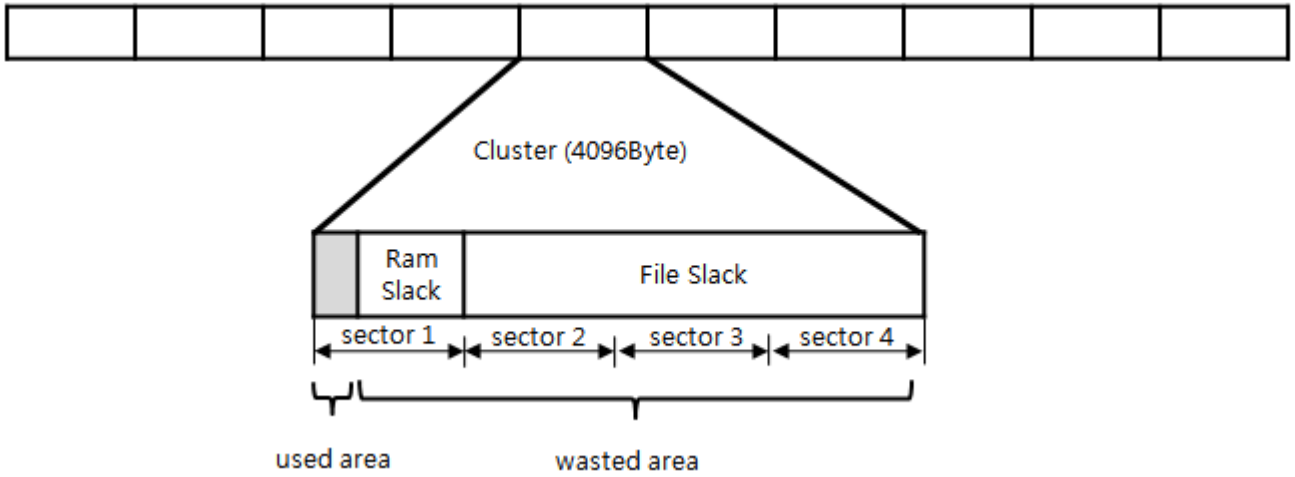
* 하드 용량 = (헤드수) * (실린더 수) * (섹터수) * (섹터당 기록 용량)

- 트랙 (Track): 디스크의 기록 단위의 하나로 자기 매체에 늘어선 동심원
- 섹터 (Sector): 트랙을 데이터 기록을 위해 나눈 가장 기본 단위



파일 시스템의 이해 – 클러스터

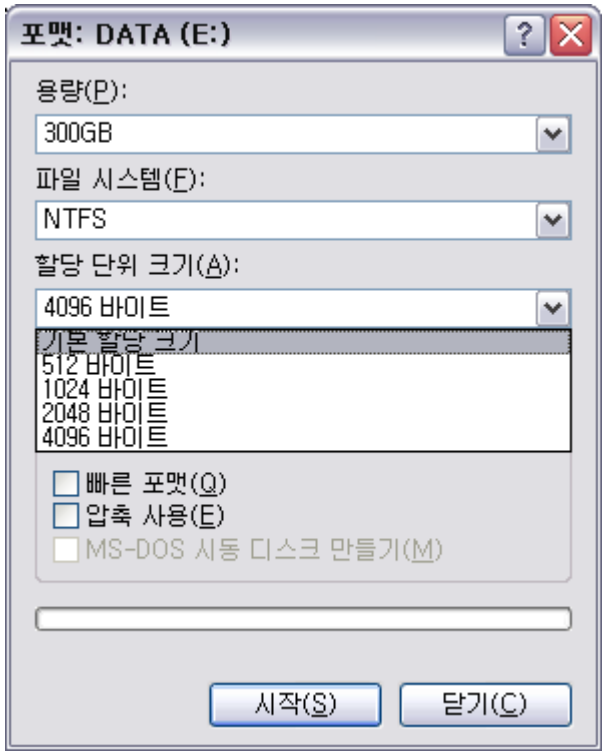
- **Cluster (클러스터)**
  - 클러스터 = 여러 개의 섹터(하드디스크의 물리적 최소 단위)를 묶은 단위
  - 섹터단위로 입출력 처리하면 시간이 오래 걸리므로 여러 개의 섹터를 묶어 한번에 처리



  - 클러스터 크기를 4,096바이트(4KB)로 지정 했을 때, 100바이트의 데이터를 저장하는 경우로 클러스터의 크기만큼 할당됨
    - 3996바이트가 낭비되지만 그럼에도 불구하고 디스크 입출력 횟수를 줄이기 위해 클러스터 단위를 사용
    - 4MB(4,096KB)파일 저장할 때 4KB크기의 클러스터 사용 = 1,024번 입출력 수행
    - 4MB(4,096KB = 4,194,304B)파일 저장할 때 512B크기의 클러스터 사용 = 8,192번 입출력 수행

파일 시스템의 이해 – 클러스터

- **Cluster (클러스터)**
  - 윈도우 시스템에서 디스크 포맷할 때 클러스터 크기 지정



FAT32에서의 클러스터 크기

볼륨 크기	클러스터 크기
32MB – 8GB	4KB
8GB – 16GB	8KB
16GB – 32GB	16KB
32GB	32KB

NTFS에서의 클러스터 크기

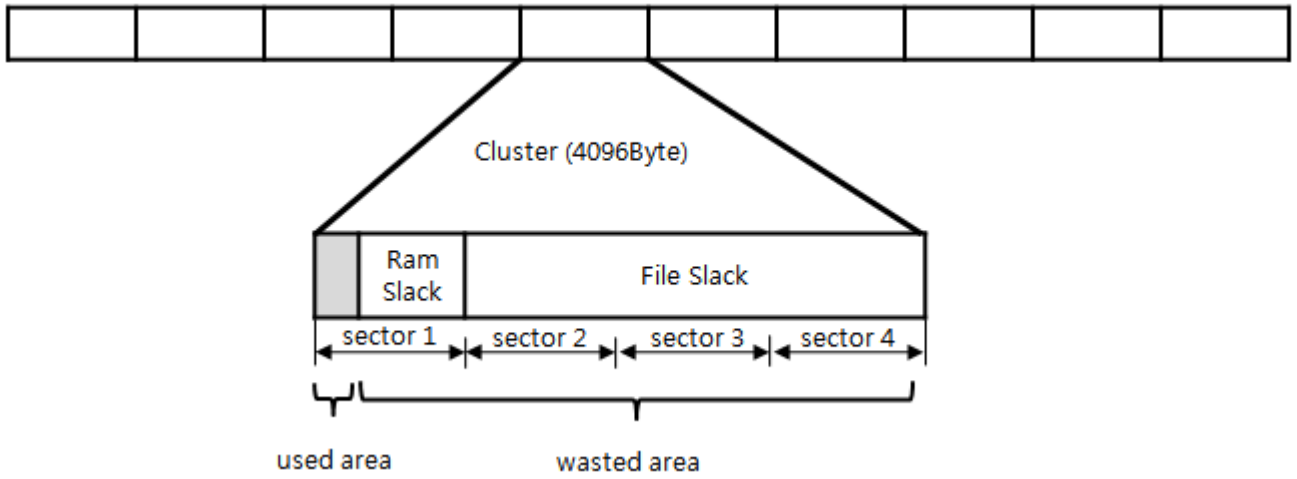
볼륨 크기	클러스터 크기
512MB 이하	512Byte
513MB – 1GB	1KB
1GB – 2GB	2KB
2GB 이상	4KB

파일 시스템의 이해 – 슬랙 공간

- **Slack Space**
  - 물리적인 구조와 논리적인 구조의 차이로 발생하는 낭비 공간
  - 물리적으로 파일에 할당된 공간이지만 논리적으로 사용 할 수 없는 낭비 공간
  - 디지털 포렌식 관점에서 - 정보 은닉 가능성, 파일 복구와 삭제된 파일의 파편 조사
    - RAM Slack (Sector Slack)
    - File Slack (Drive Slack)
      - 이전에 사용한 데이터가 존재, 흔적 조사에 활용
    - File System Slack
    - Volume Slack

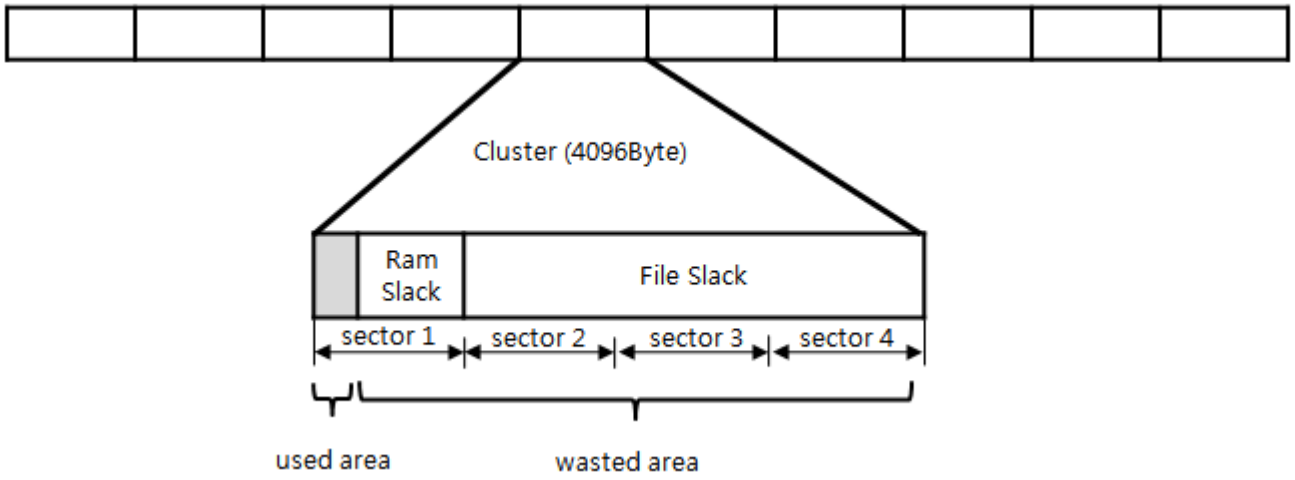
파일 시스템의 이해 – 슬랙 공간

- **Slack Space (RAM Slack & File Slack)**
  - RAM Slack (Sector Slack)
    - 램에 저장 되어있는 데이터가 디스크에 저장될 때 512 바이트씩 기록되는 특성 때문에 발생하는 공간으로 섹터 슬랙(Sector Slack)이라고도 함
    - 지정되는 파일 크기가 512 바이트의 배수가 아닐 경우 발생
    - 여분 바이트 0x00 값으로 기록
    - 램 슬랙을 이용하면 파일의 끝을 알 수 있기 때문에 삭제된 파일 복구 시 유용하게 사용



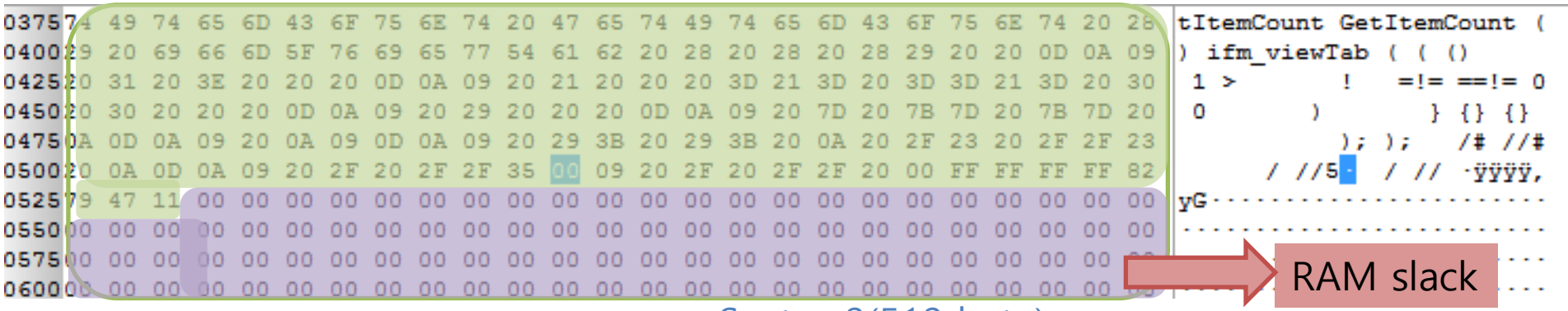
파일 시스템의 이해 – 슬랙 공간

- **Slack Space (RAM Slack & File Slack)**
  - File Slack (Drive Slack)
    - 클러스터의 사용으로 인해 낭비되는 공간 중 램 슬랙을 제외한 부분으로 드라이브 슬랙(Drive Slack)이라고도 함
    - 파일 슬랙을 이용하면 특정 파일이 해당 저장 매체에 존재하였는지 규명 가능
      - 존재 여부를 알아야 할 파일을 클러스터 단위로 나눈 후, 각 클러스터의 마지막 부분과 파일 슬랙 중 일치하는 부분이 있는지 확인
    - 최하단의 디스크 입출력은 섹터 단위로 진행되므로 0x00으로 기록되는 램 슬랙과 다르게 이전의 데이터가 그대로 남아있음(I/O는 섹터단위로 진행)

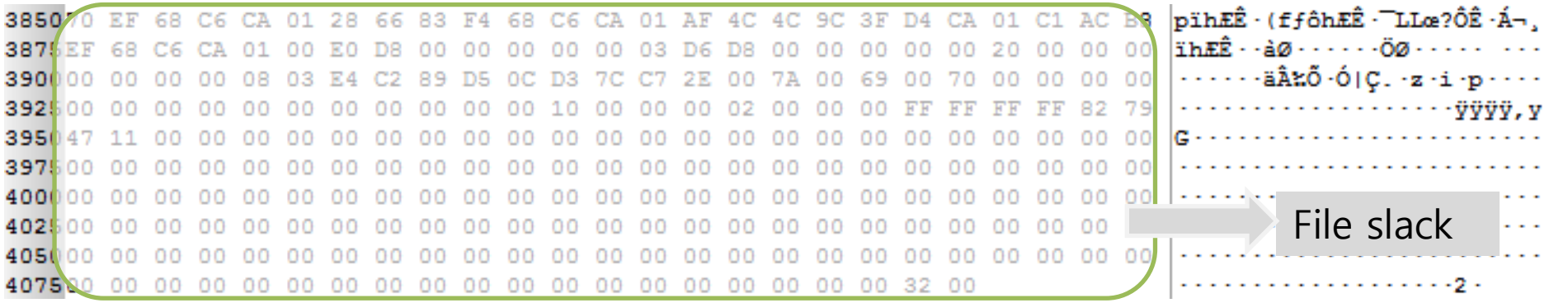


파일 시스템의 이해 – 슬랙 공간

Sector 1(512 byte)



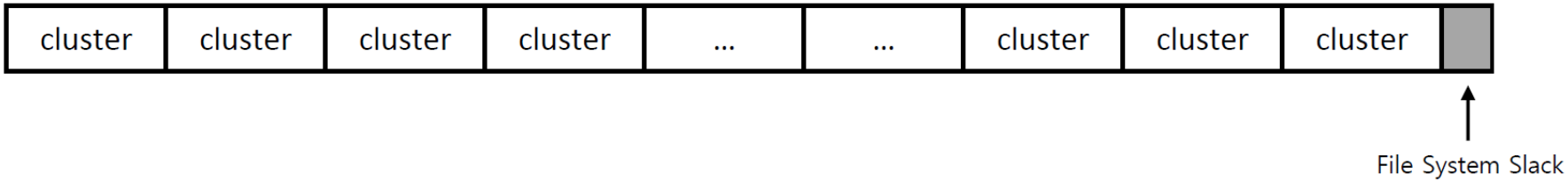
Sector 2(512 byte)



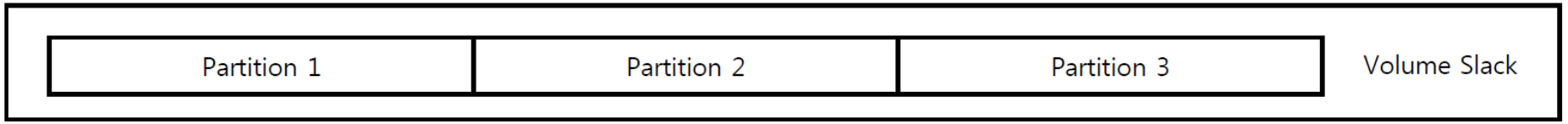
Sector 8(512 byte)

파일 시스템의 이해 – 슬랙 공간

- **Slack Space (RAM Slack & File Slack)**
  - File System Slack
    - ✓ 파일시스템 할당 크기와 볼륨 크기간의 차이로 인해 발생되는 공간
      - 1,026KB 볼륨에 4KB 클러스터 사용하는 파일 시스템 구성하면 마지막 2KB이 파일 시스템 슬랙이 됨

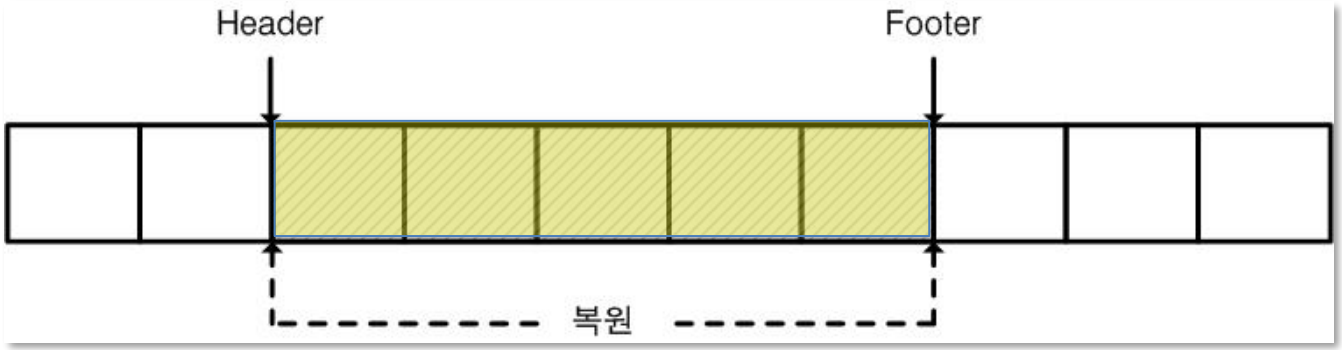


  - Volume Slack
    - ✓ 전체 볼륨 크기와 할당된 파티션 크기의 차이로 인해 발생되는 공간



파일 복구 – 파일 시스템 상의 파일 복구

- **파일 시스템 정보를 얻을 수 없는 경우의 복구**
  - 파일 시스템에서 얻을 수 있는 정보 없이 '파일 자체 정보' 기반 복구
    즉, 파일의 고유한 특성이 있는 파일만 복구 가능
  - 연속적으로 존재하는 파일에 대한 복구는 대부분 가능, 조각난 경우는 어려움
  - 추출된 파일이 올바른 파일이라는 보장이 없음
  - 많은 시간이 소요됨



파일 복구 – 파일 시스템 상의 파일 복구

- **파일이 연속적이지 않고 조각난 경우**
  - 조각난 파일이 생성되는 이유
    - ✓ 파일을 저장할 충분한 연속 공간이 없을 경우
    - ✓ 기존 파일에 데이터가 추가될 때, 파일의 후반부 영역에 할당되지 않은 영역의 크기가 충분하지 않은 경우
  - 파일 포맷의 특성을 이용 복구
  - Pattern Recognition, 통계 분석 등

파일 복구 - 파일 카빙

- **파일 카빙(File Carving) 기법 : 파일시스템의 도움없이 저장 매체의 비할당 영역으로부터 파일을 복구하는 기법**

  ✓ **연속적인 카빙 기법**

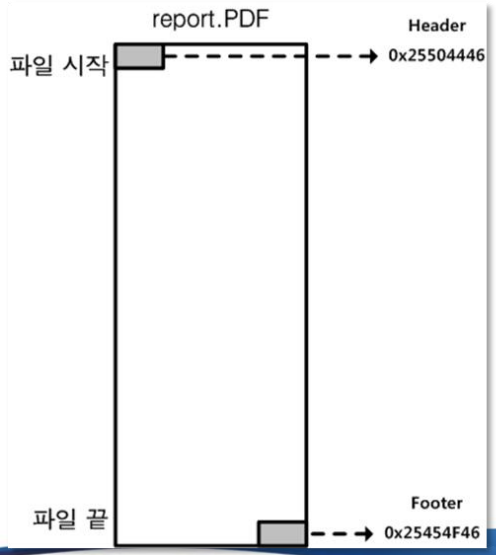
  ➢ 파일 내용이 저장 매체의 연속된 공간에 저장된 경우 수행

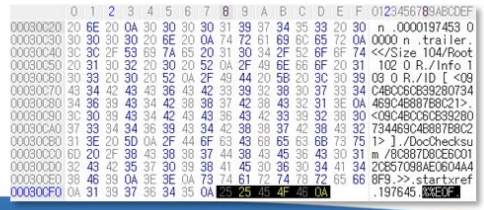
  ✓ **비연속적인 카빙 기법**

  ➢ 파일의 내용이 저장 매체의 여러 부분에 조각나 저장된 경우 수행

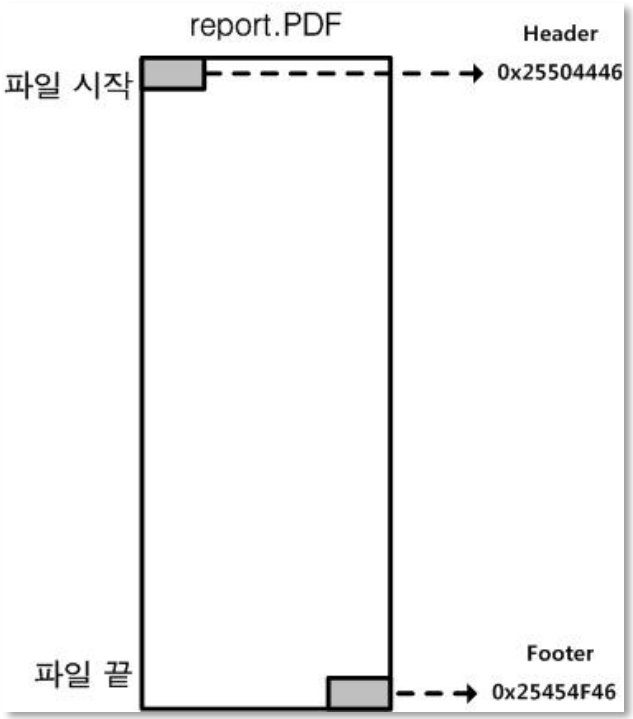
파일 복구 - 파일 카빙

- **시그니처 (Signature) 기반 카빙**
  - 파일 포맷별로 존재하는 고유한 시그니처를 이용하는 방법
  - 헤더(Header)와 푸터(Footer) 시그니처가 모두 존재하는 파일의 경우
    두 시그니처 사이의 데이터가 파일의 내용
  - 파일의 Header와 Footer 정보
    - 일부 파일은 파일의 시작과 끝을 알 수 있는 고유한 Header와 Footer를 가짐
    - PDF, GIF. PNG, JPG, ALZ, ZIP, RAR, MPG …





파일 복구 - 파일 카빙



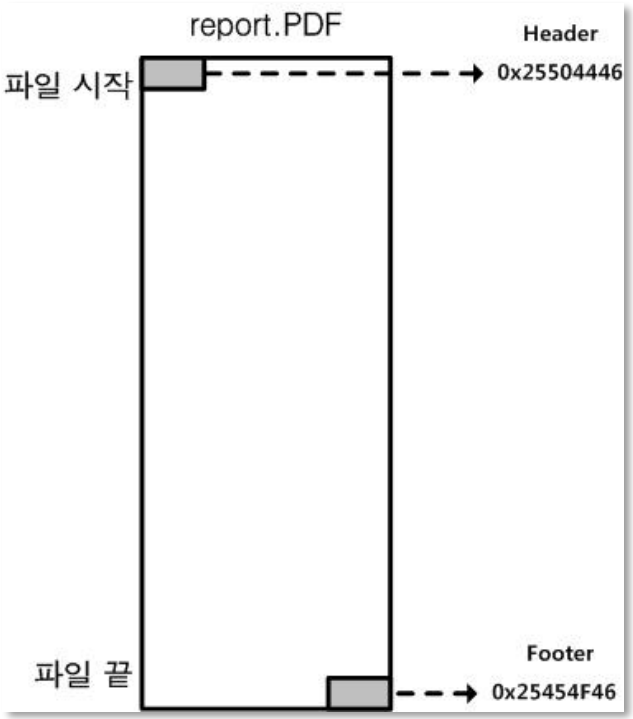
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF
00000000	25	50	44	46	2D	31	2E	34	0A	25	C3	A4	C3	BC	C3	B6	%PDF-1.4.%.....
00000010	C3	9F	0A	32	20	30	20	6F	62	6A	0A	3C	3C	2F	4C	65	...2 0 obj.<</Le
00000020	6E	67	74	68	20	33	20	30	20	52	2F	46	69	6C	74	65	ngth 3 0 R/Filte
00000030	72	2F	46	6C	61	74	65	44	65	63	6F	64	65	3E	3E	0A	r/FlateDecode>>.
00000040	73	74	72	65	61	6D	0A	78	9C	95	56	4B	6B	DC	30	10	stream.x..VKk.O.
00000050	BE	EF	AF	D0	B9	B0	AE	66	F4	B2	41	18	76	9D	DD	43	.......f..A.v..C
00000060	6F	81	85	1E	4A	6F	6D	DA	43	5A	68	2E	FD	FB	9D	87	o...Jom.CZh.....
00000070	64	CB	8E	37	A4	04	B4	8A	34	CF	6F	BE	19	D9	76	60	d..7....4.o...v`
00000080	FE	1E	FE	18	6B	8E	B6	73	26	0D	AE	EB	4D	18	02	ED	....k..s&...M...
00000090	5F	BE	9B	CF	1F	CC	EF	83	ED	42	DF	A3	33	B6	B3	43	_........B..3..C
000000A0	6F	F9	17	30	F8	C1	BC	FC	38	40	AF	6A	BF	0E	45	FF	o..0....8@.j..E.
000000B0	99	0C	91	36	FD	BA	B8	6C	CA	55	95	7E	36	3F	0F	4F	...6...l.U.~6?.O
000000C0	1F	C4	29	FF	91	9D	F3	ED	E0	1D	45	E2	87	2E	9A	DB	..)........E.....
000000D0	37	F3	F1	0A	06	D0	DC	9E	BE	64	0B	23	66	8B	E3	91	7........d.#f...

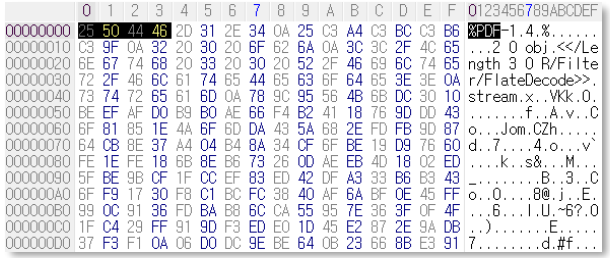
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF
00030C20	20	6E	20	0A	30	30	30	30	31	39	37	34	35	33	20	30	n .0000197453 0
00030C30	30	30	30	30	20	6E	20	0A	74	72	61	69	6C	65	72	0A	0000 n .trailer.
00030C40	3C	3C	2F	53	69	7A	65	20	31	30	34	2F	52	6F	6F	74	<</Size 104/Root
00030C50	20	31	30	32	20	30	20	52	0A	2F	49	6E	66	6F	20	31	102 0 R./Info 1
00030C60	30	33	20	30	20	52	0A	2F	49	44	20	5B	20	3C	30	39	03 0 R./ID [<09
00030C70	43	34	42	43	43	36	43	42	33	39	32	38	30	37	33	34	C4BCC6CB39280734
00030C80	34	36	39	43	34	42	38	38	37	42	38	43	32	31	3E	0A	469C4B887B8C21>.
00030C90	3C	30	39	43	34	42	43	43	36	43	42	33	39	32	38	30	<09C4BCC6CB39280
00030CA0	37	33	34	34	36	39	43	34	42	38	38	37	42	38	43	32	734469C4B887B8C2
00030CB0	31	3E	20	5D	0A	2F	44	6F	63	43	68	65	63	6B	73	75	1>]./DocChecksu
00030CC0	6D	20	2F	38	43	38	38	37	44	38	43	45	36	43	30	31	m /8C887D8CE6C01
00030CD0	32	43	42	35	37	30	39	38	41	45	30	36	30	34	41	34	2CB57098AE0604A4
00030CE0	38	46	39	0A	3E	3E	0A	73	74	61	72	74	78	72	65	66	8F9.>>.startxref
00030CF0	0A	31	39	37	36	34	35	0A	25	25	45	4F	46	0A			.197645.%%EOF.

파일 복구 - 파일 카빙

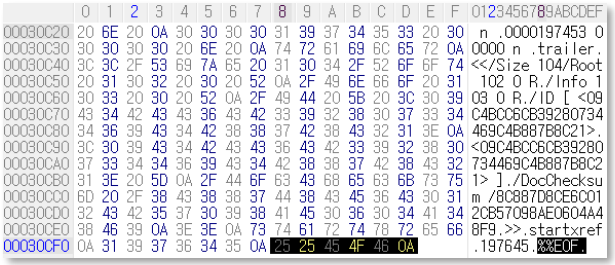
파일의 Header와 Footer 정보

- 일부 파일은 파일의 시작과 끝을 알 수 있는 고유한 Header와 Footer를 가짐
  - PDF, GIF. PNG, JPG, ALZ, ZIP, RAR, MPG …





	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF
00000000	25	50	44	46	2D	31	2E	34	0A	25	C3	A4	C3	BC	C3	B6	%PDF-1.4.%.....
00000010	C3	9F	0A	32	20	30	20	6F	62	6A	0A	3C	3C	2F	4C	65	...2 0 obj.<</Le
00000020	6E	67	74	68	20	33	20	30	20	52	2F	46	69	6C	74	65	ngth 3 0 R/Filte
00000030	72	2F	46	6C	61	74	65	44	65	63	6F	64	65	3E	3E	0A	r/FlateDecode>>.
00000040	73	74	72	65	61	6D	0A	78	9C	95	56	4B	6B	DC	30	10	stream.x..VKk.0.
00000050	BE	EF	AF	D0	B9	B0	AE	66	F4	B2	41	18	76	9D	DD	43	.......f..A.v..C
00000060	6F	81	85	1E	4A	6F	6D	DA	43	5A	68	2E	FD	FB	9D	87	o...Jom.CZh.....
00000070	64	CB	8E	37	A4	04	B4	8A	34	CF	6F	BE	19	D9	76	60	d..7....4.o...v`
00000080	FE	1E	FE	18	6B	8E	B6	73	26	0D	AE	EB	4D	18	02	ED	....k..s&...M...
00000090	5F	BE	9B	CF	1F	CC	EF	83	ED	42	DF	A3	33	B6	B3	43	_........B..3..C
000000A0	6F	F9	17	30	F8	C1	BC	FC	38	40	AF	6A	BF	0E	45	FF	o..0....8@.j..E.
000000B0	99	0C	91	36	FD	BA	B8	6C	CA	55	95	7E	36	3F	0F	4F	...6...l.U.~6?.O
000000C0	1F	C4	29	FF	91	9D	F3	ED	E0	1D	45	E2	87	2E	9A	DB	..)........E.....
000000D0	37	F3	F1	0A	06	D0	DC	9E	BE	64	0B	23	66	8B	E3	91	7........d.#f...



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF
00030C20	20	6E	20	0A	30	30	30	30	31	39	37	34	35	33	20	30	n .0000197453 0
00030C30	30	30	30	30	20	6E	20	0A	74	72	61	69	6C	65	72	0A	0000 n .trailer.
00030C40	3C	3C	2F	53	69	7A	65	20	31	30	34	2F	52	6F	6F	74	<</Size 104/Root
00030C50	20	31	30	32	20	30	20	52	0A	2F	49	6E	66	6F	20	31	102 0 R./Info 1
00030C60	30	33	20	30	20	52	0A	2F	49	44	20	5B	20	3C	30	39	03 0 R./ID [<09
00030C70	43	34	42	43	43	36	43	42	33	39	32	38	30	37	33	34	C4BCC6CB39280734
00030C80	34	36	39	43	34	42	38	38	37	42	38	43	32	31	3E	0A	469C4B887B8C21>.
00030C90	3C	30	39	43	34	42	43	43	36	43	42	33	39	32	38	30	<09C4BCC6CB39280
00030CA0	37	33	34	34	36	39	43	34	42	38	38	37	42	38	43	32	734469C4B887B8C2
00030CB0	31	3E	20	5D	0A	2F	44	6F	63	43	68	65	63	6B	73	75	1>]./DocChecksu
00030CC0	6D	20	2F	38	43	38	38	37	44	38	43	45	36	43	30	31	m /8C887D8CE6C01
00030CD0	32	43	42	35	37	30	39	38	41	45	30	36	30	34	41	34	2CB57098AE0604A4
00030CE0	38	46	39	0A	3E	3E	0A	73	74	61	72	74	78	72	65	66	8F9.>>.startxref
00030CF0	0A	31	39	37	36	34	35	0A	25	25	45	4F	46	0A			.197645.%%EOF.

파일 복구 - 파일 카빙

- **램 슬랙 카빙**

✓ 푸터 시그니처 이후에 램 슬랙이 존재

✓ 시그니처 기반 카빙 기법에서 푸터 시그니처와 함께 확인하여 많은 오탐을 줄일 수 있음

파일 복구 - 파일 카빙

파일 구조체 카빙

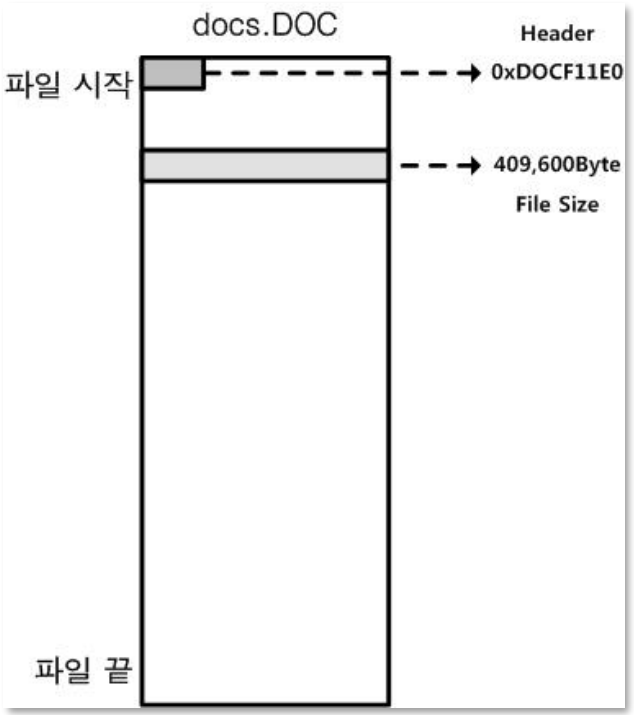
✓ 푸터 시그니처가 존재하지 않거나 파일 포맷 내부에 여러 개의 시그니처가 존재하는 경우 효과적인 방법으로 파일의 구조를 분석하여 카빙 하는 기법

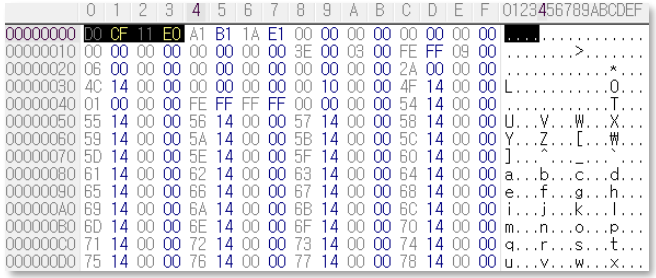
- 파일 크기 획득 방법
- 파일 구조 검증 방법

파일 복구 - 파일 카빙(파일 크기 획득 방법)

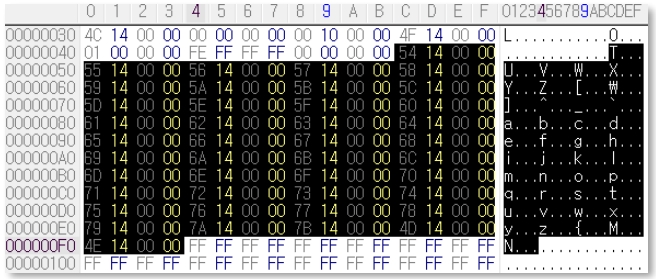
파일의 Header와 File size 정보

- 일부 파일은 파일의 시작과 크기를 알 수 있는 정보를 포함하고 있음
  - DOC, ODT, ODS, BMP, AVI, ASF, WAV ...





	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF
00000000	D0	CF	11	E0	A1	B1	1A	E1	00	00	00	00	00	00	00	00	....á±.á.....
00000010	00	00	00	00	00	00	00	00	3E	00	03	00	FE	FF	09	00	........>.....
00000020	06	00	00	00	00	00	00	00	00	00	00	00	2A	00	00	00	..........*...
00000030	4C	14	00	00	00	00	00	00	00	10	00	00	4F	14	00	00	L.........O...
00000040	01	00	00	00	FE	FF	FF	FF	00	00	00	00	54	14	00	00	............T...
00000050	55	14	00	00	56	14	00	00	57	14	00	00	58	14	00	00	U...V...W...X...
00000060	59	14	00	00	5A	14	00	00	5B	14	00	00	5C	14	00	00	Y...Z...[...₩...
00000070	5D	14	00	00	5E	14	00	00	5F	14	00	00	60	14	00	00	]...^..._...`...
00000080	61	14	00	00	62	14	00	00	63	14	00	00	64	14	00	00	a...b...c...d...
00000090	65	14	00	00	66	14	00	00	67	14	00	00	68	14	00	00	e...f...g...h...
000000A0	69	14	00	00	6A	14	00	00	6B	14	00	00	6C	14	00	00	i...j...k...l...
000000B0	6D	14	00	00	6E	14	00	00	6F	14	00	00	70	14	00	00	m...n...o...p...
000000C0	71	14	00	00	72	14	00	00	73	14	00	00	74	14	00	00	q...r...s...t...
000000D0	75	14	00	00	76	14	00	00	77	14	00	00	78	14	00	00	u...v...w...x...



	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	0123456789ABCDEF
00000030	4C	14	00	00	00	00	00	00	00	10	00	00	4F	14	00	00	L.........O...
00000040	01	00	00	00	FE	FF	FF	FF	00	00	00	00	54	14	00	00	............T...
00000050	55	14	00	00	56	14	00	00	57	14	00	00	58	14	00	00	U...V...W...X...
00000060	59	14	00	00	5A	14	00	00	5B	14	00	00	5C	14	00	00	Y...Z...[...₩...
00000070	5D	14	00	00	5E	14	00	00	5F	14	00	00	60	14	00	00	]...^..._...`...
00000080	61	14	00	00	62	14	00	00	63	14	00	00	64	14	00	00	a...b...c...d...
00000090	65	14	00	00	66	14	00	00	67	14	00	00	68	14	00	00	e...f...g...h...
000000A0	69	14	00	00	6A	14	00	00	6B	14	00	00	6C	14	00	00	i...j...k...l...
000000B0	6D	14	00	00	6E	14	00	00	6F	14	00	00	70	14	00	00	m...n...o...p...
000000C0	71	14	00	00	72	14	00	00	73	14	00	00	74	14	00	00	q...r...s...t...
000000D0	75	14	00	00	76	14	00	00	77	14	00	00	78	14	00	00	u...v...w...x...
000000E0	79	14	00	00	7A	14	00	00	7B	14	00	00	4D	14	00	00	y...z...{...M...
000000F0	4E	14	00	00	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	N..............
00000100	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	FF	....

파일 복구 - 파일 카빙(파일 구조 검증 방법)

- ✓ 데이터 표현 위한 고유한 계층 구조를 검증하여 카빙하는 방법
- ✓ 문서 파일과 같이 빈번한 수정이 이루어지는 경우 사용

파일 구조 검증 방법을 적용할 수 있는 파일 포멧

파일 포맷	시그니처	
	헤더 (Hex)	푸터 (Hex)
ZIP	50 4B 03 04	50 4B 05 06
ALZ	41 4C 5A 01	43 4C 5A 02
RAR	52 61 72 21 1A 07	3D 7B 00 40 07 00
Compound	D0 CF 11 E0 A1 B1 1A E1	-

Linux 실습 싸이트 참고

- **https://bellard.org/jslinux**

참고문헌

- 대검찰정 과학수사본부 : 디지털포렌식센터
  - https://www.spo.go.kr/spo/major/forensics/forensics01.jsp
- 경찰청 사이버 안전국 : 디지털포렌식센터
  - http://cyber.go.kr/bureau/sub4.jsp?mid=040401
- 사이버포렌식협회
  - http://www.cfpa.or.kr/intro2.htm
- 한국포렌식학회
  - https://kdfs.jams.or.kr/co/main/jmMain.kci
- 한국디지털포렌식전문가협회
  - http://fka.kr/
- 하드디스크(Hard Disk Drive, HDD) 구조와 작동 원리 및 각종 규격, https://whitesnake1004.tistory.com/273

Q & A