HFS+ File system

SangJun Jeon

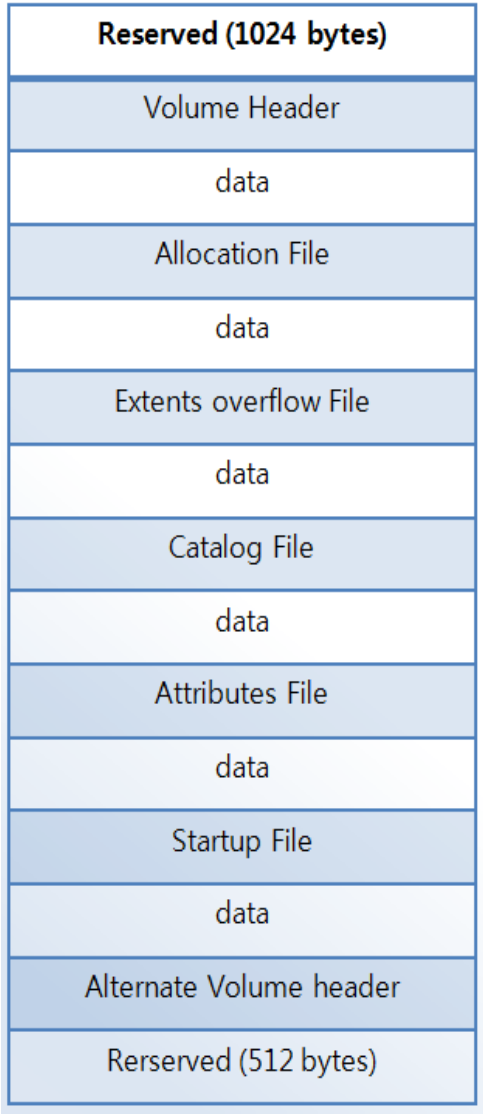
heros86@korea.ac.kr

DFRC@Korea University

HFS+ File System

HFS+ File System

- **HFS Filesystem**
  - UFS 기반으로 제작한 파일 시스템
  - 저널링을 제공하지 않음
  - 파일 이름 길이 제한(255자)
  - 큰 용량 데이터 처리 문제가 존재
- **HFS+ Filesystem**
  - Mac OS X를 위해 개발한 파일시스템
  - 디스크 및 CD-ROM에서도 사용할 수 있도록 구성
  - HFS 파일시스템의 단점을 보완



Raw 이미지 획득

- **MAC OS X는 Target Disk Mode를 가지고 있음**
  - FireWire로 두 시스템을 연결
  - Mac OS X의 auto mount 데몬을 종료하고 수행
    - /usr/sbin/diskarbitrationd
  - 타겟 시스템을 T를 누른 채로 부팅
    - Target Disk Mode
    - Support to Mac OS X or OS8/OS9
  - FireWire를 이용한 디스크 이미징

Raw 이미지 획득

- **Net cat 을 이용한 DD 이미지 전송**
  - XP (서버측 설정)
    - 12345 포트 Open
    - \>nc -w 10 -Lvp 12345 > Mac.dd
  - Mac (Client측 명령)
    - $sudo dd if=/dev/disk1 bs=1024 | nc 163.152.165.109 12345

```
SANG-JUN-JEONui-Mac:/ sangjunjeon$ sudo dd if=/dev/disk1 bs=4096 | nc 163.152.16
5.109 12345
Password:
3932160+0 records in
3932160+0 records out
16106127360 bytes transferred in 3779.594762 secs (4261337 bytes/sec)
SANG-JUN-JEONui-Mac:/ sangjunjeon$
```

Raw 이미지 획득

- **이미지 정상 여부 판단**
  - Sleuthkit 3.1.0 버전 hfs 지원

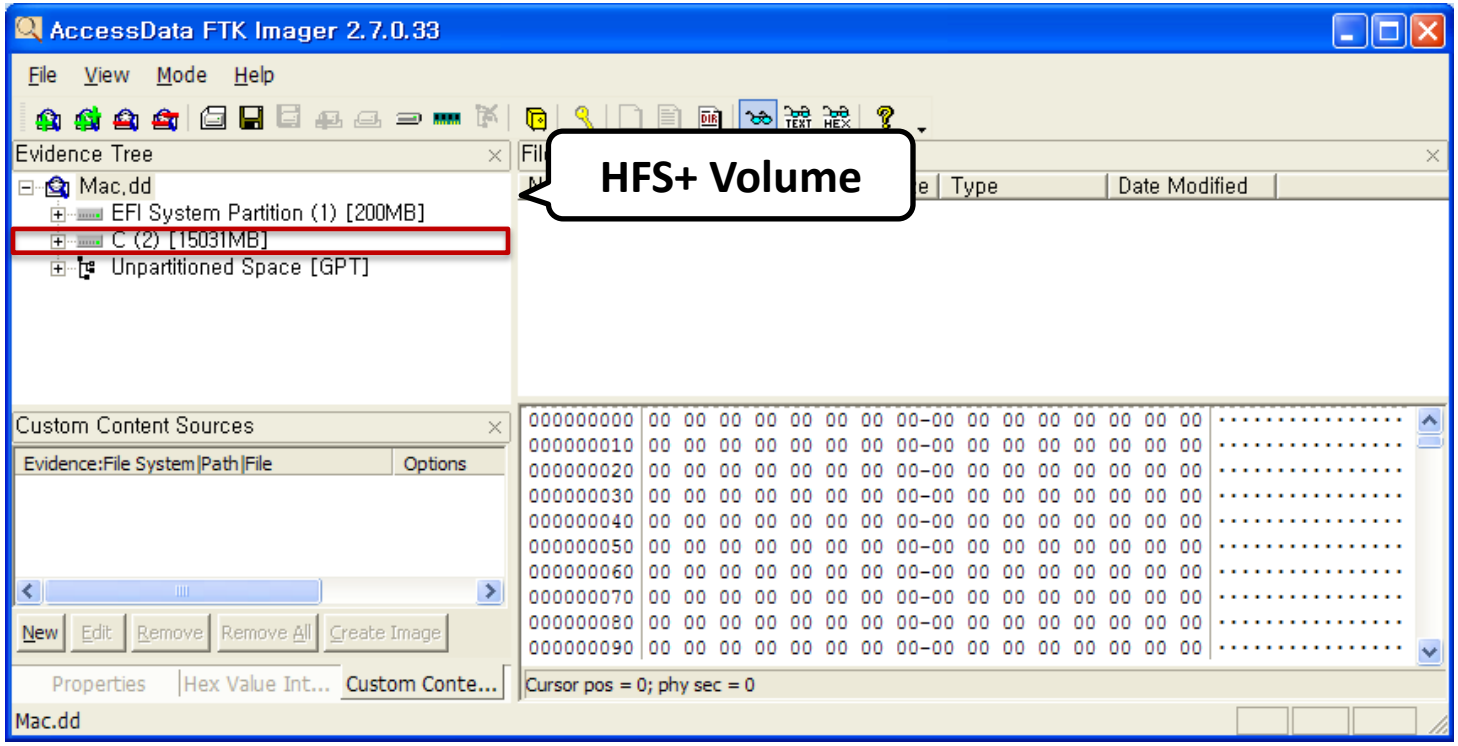
```
C:₩sleuthkit>fsstat.exe -f list
Supported file system types:
        ntfs (NTFS)
        fat (FAT (Auto Detection))
        ext (ExtX (Auto Detection))
        iso9660 (ISO9660 CD)
        hfs (HFS+)
        ufs (UFS (Auto Detection))
        raw (Raw Data)
        swap (Swap Space)
        fat12 (FAT12)
        fat16 (FAT16)
        fat32 (FAT32)
        ext2 (Ext2)
        ext3 (Ext3)
        ufs1 (UFS1)
        ufs2 (UFS2)
```

  - 올바른 이미지가 확보되었는지 판단하기 위해 sleuth kit 이용.
    - 인식 에러
    - 볼륨이 아닌 disk를 이미징 한 결과

```
C:₩sleuthkit>fsstat.exe "f:₩Mac 연동자료₩Mac.dd"
Cannot determine file system type
```

Raw 이미지 획득

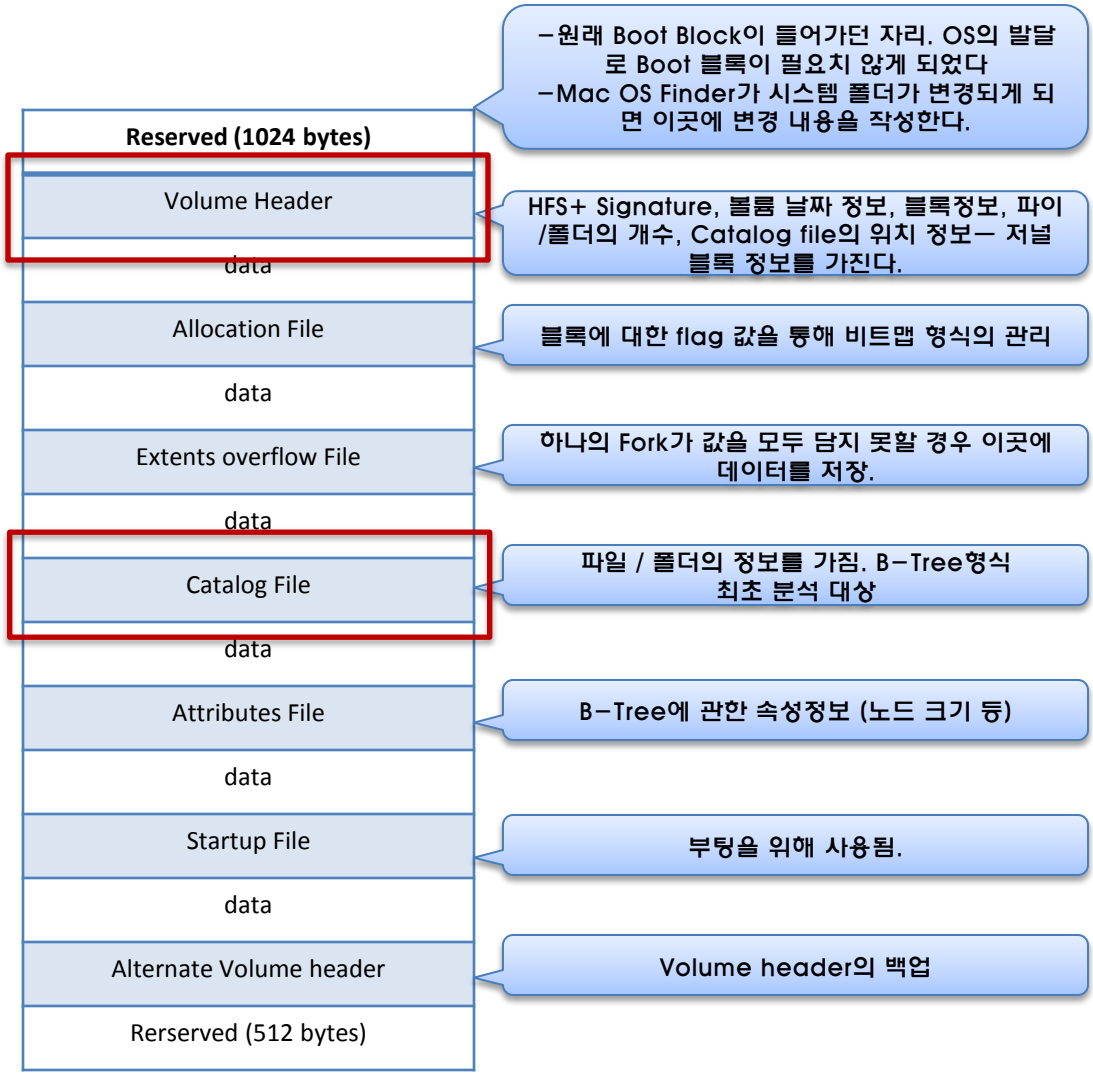
- **EFI Disk label 제거**
  - FTK Imager등을 이용, HFS+ Volume만 이미징



Basic Structure

HFS+ 구조

- 기본구조


Reserved (1024 bytes)
Volume Header
data
Allocation File
data
Extents overflow File
data
Catalog File
data
Attributes File
data
Startup File
data
Alternate Volume header
Rerserved (512 bytes)
-원래 Boot Block이 들어가던 자리. OS의 발달로 Boot 블록이 필요치 않게 되었다
-Mac OS Finder가 시스템 폴더가 변경되게 되면 이곳에 변경 내용을 작성한다.
HFS+ Signature, 볼륨 날짜 정보, 블록정보, 파이/폴더의 개수, Catalog file의 위치 정보— 저널 블록 정보를 가진다.
블록에 대한 flag 값을 통해 비트맵 형식의 관리
하나의 Fork가 값을 모두 담지 못할 경우 이곳에 데이터를 저장.
파일 / 폴더의 정보를 가짐. B-Tree형식 최초 분석 대상
B-Tree에 관한 속성정보 (노드 크기 등)
부팅을 위해 사용됨.
Volume header의 백업

HFS+ 구조

- **Volume Header**
  - struct HFSPlusVolumeHeader {

```
UInt16 signature;              //2byte
UInt16 version;                //2byte
UInt32 attributes;             //4byte
UInt32 lastMountedVersion;     //4byte
UInt32 journalInfoBlock;       //4byte
UInt32 createDate;             //4byte
UInt32 modifyDate;             //4byte
UInt32 backupDate;             //4byte
UInt32 checkedDate;            //4byte
UInt32 fileCount;              //4byte
UInt32 folderCount;            //4byte
UInt32 blockSize;              //4byte
UInt32 totalBlocks;            //4byte
UInt32 freeBlocks;             //4byte
UInt32 nextAllocation;         //4byte
UInt32 rsrcClumpSize;          //4byte
UInt32 dataClumpSize;          //4byte
HFSCatalogNodeID nextCatalogID;//4byte
UInt32 writeCount;             //4byte
UInt64 encodingsBitmap;        //8byte
UInt32 finderInfo[8];          //32byte
HFSPlusForkData allocationFile;//80byte
HFSPlusForkData extentsFile;   //80byte
HFSPlusForkData catalogFile;   //80byte
HFSPlusForkData attributesFile;//80byte
HFSPlusForkData startupFile;   //80byte
```

}; typedef struct HFSPlusVolumeHeader HFSPlusVolumeHeader;

HFS+ 구조

- **HFSPlusForkData 구조체**

```
struct HFSPlusForkData {
    UInt64 logicalSize;                 //8byte
    UInt32 clumpSize;                   //4byte
    UInt32 totalBlocks;                 //4byte
    HFSPlusExtentDescriptor extents;    //64byte
}; typedef struct HFSPlusForkData HFSPlusForkData;          //80byte
```

112byte data			
Allocation File(80byte)			
Extents File(80byte)			
Logical Size (8byte)		Clump Size (4byte)	Total Blocks(4byte)
Start Block	Block Count	Start Block	Block Count
Start Block	Block Count	Start Block	Block Count
Start Block	Block Count	Start Block	Block Count
Start Block	Block Count	Start Block	Block Count
Attributes File(80byte)			
Startup File(80byte)			

HFS+ 구조

- **HFSPlusExtentDescriptor 구조체**

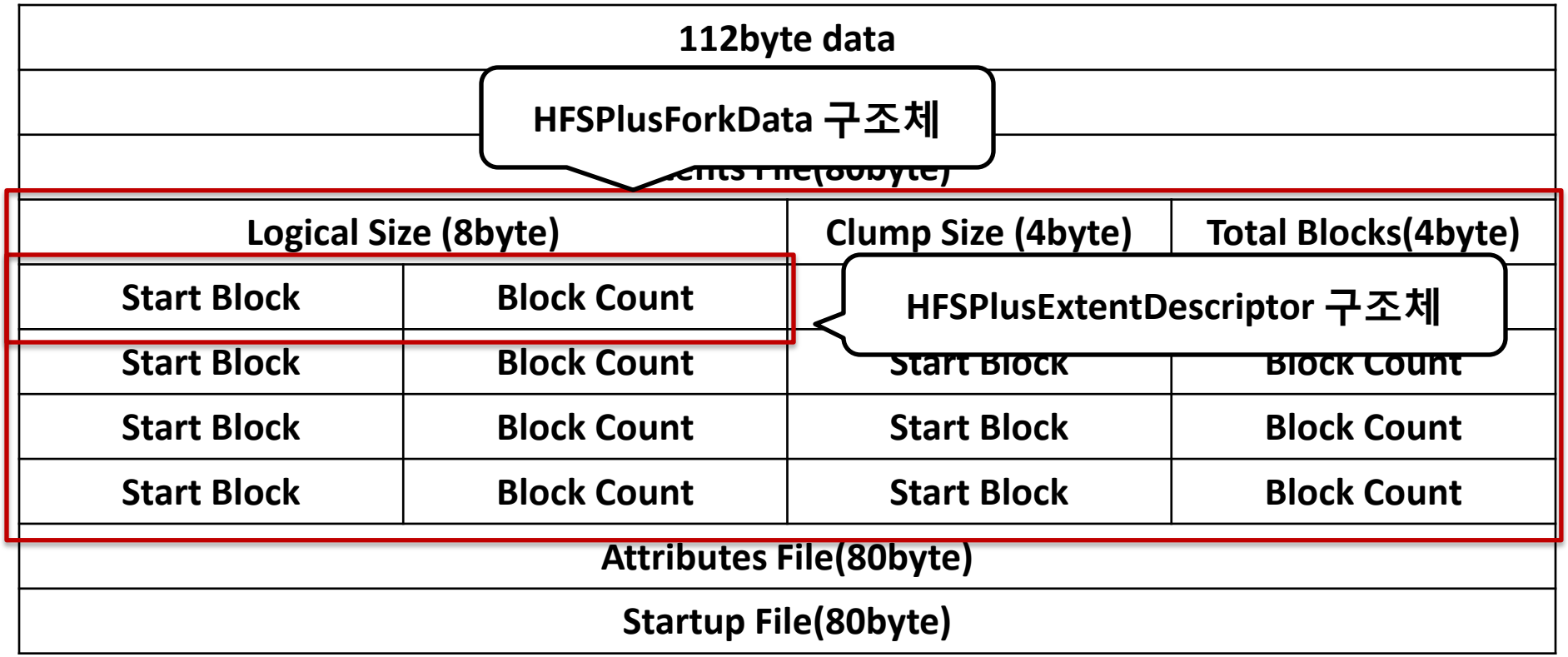
```
typedef   HFSPlusExtentDescriptor   HFSPlusExtentRecord[8];          //8byte

struct   HFSPlusExtentDescriptor {
    UInt32 startBlock;                                               //4byte
    UInt32 blockCount;                                               //4byte
}; typedef struct HFSPlusExtentDescriptor HFSPlusExtentDescriptor;
```

112byte data			
Allocation File(80byte)			
Extents File(80byte)			
Logical Size (8byte)		Clump Size (4byte)	Total Blocks(4byte)
Start Block	Block Count	Start Block	Block Count
Start Block	Block Count	Start Block	Block Count
Start Block	Block Count	Start Block	Block Count
Start Block	Block Count	Start Block	Block Count
Attributes File(80byte)			
Startup File(80byte)			

HFS+ 구조

- Volume Header



HFS+ 구조

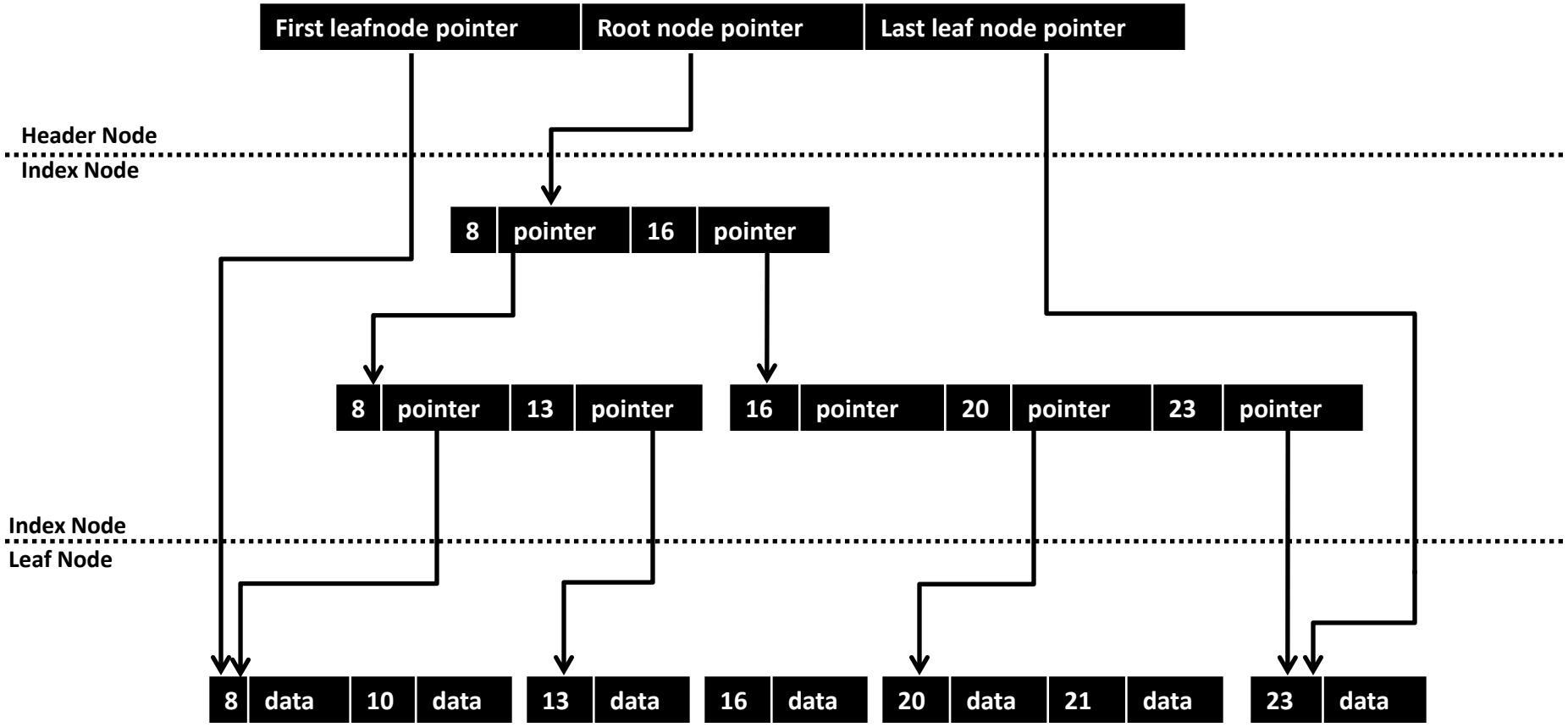
- **HFS+ Volume Header**

112byte data			
Allocation File(80byte)			
Extents File(80byte)			
Logical Size (8byte)		Clump Size (4byte)	Total Blocks(4byte)
Start Block	Block Count	Start Block	Block Count
Start Block	Block Count	Start Block	Block Count
Start Block	Block Count	Start Block	Block Count
Start Block	Block Count	Start Block	Block Count
Attributes File(80byte)			
Startup File(80byte)			

$CatalogFile 속성

HFS+ 구조

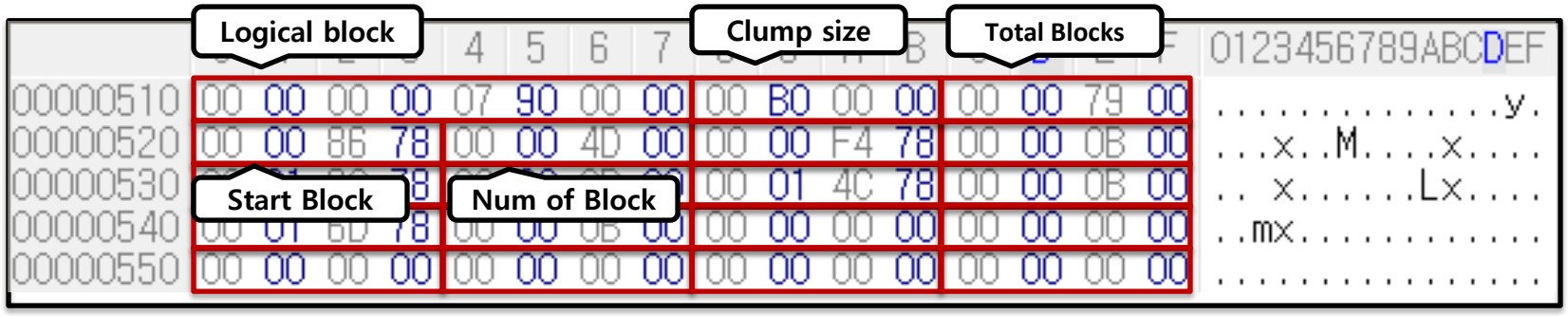
- Catalog File 기본 구조



Catalog File

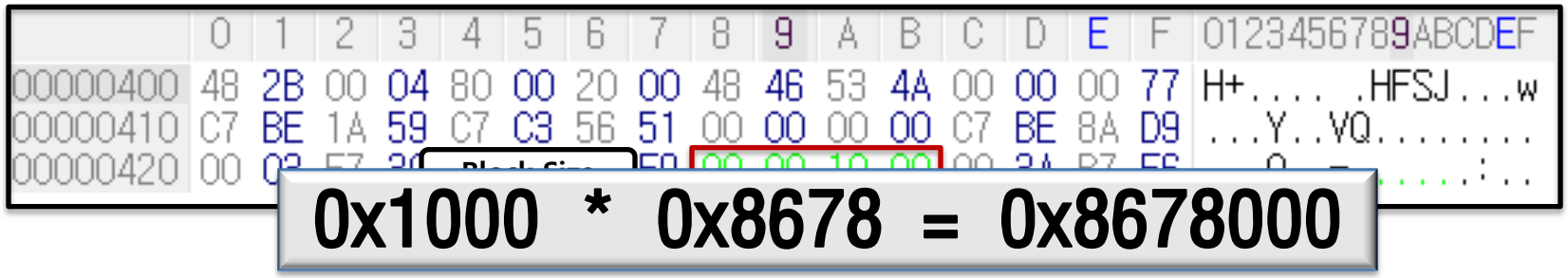
Catalog File(Find Root Node)

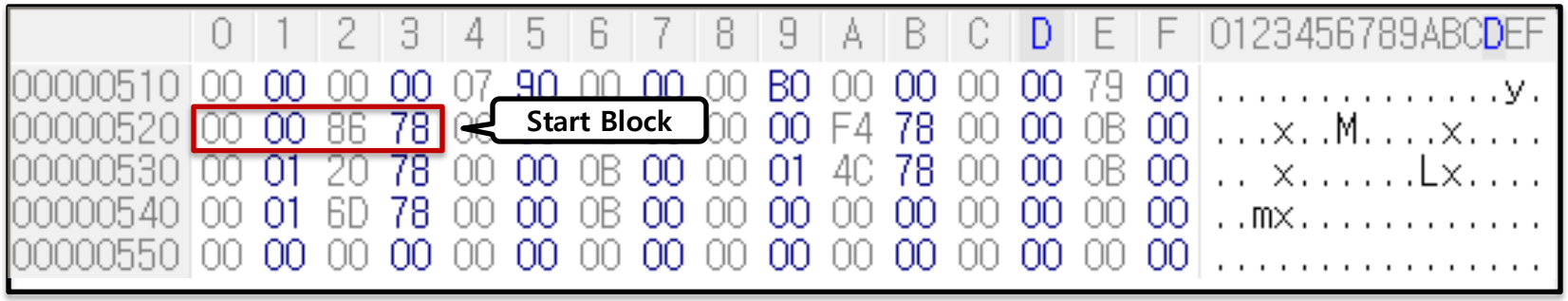
- **HFS+ Volume Header**
  - $CatalogFile 속성



Catalog File(Find Root Node)

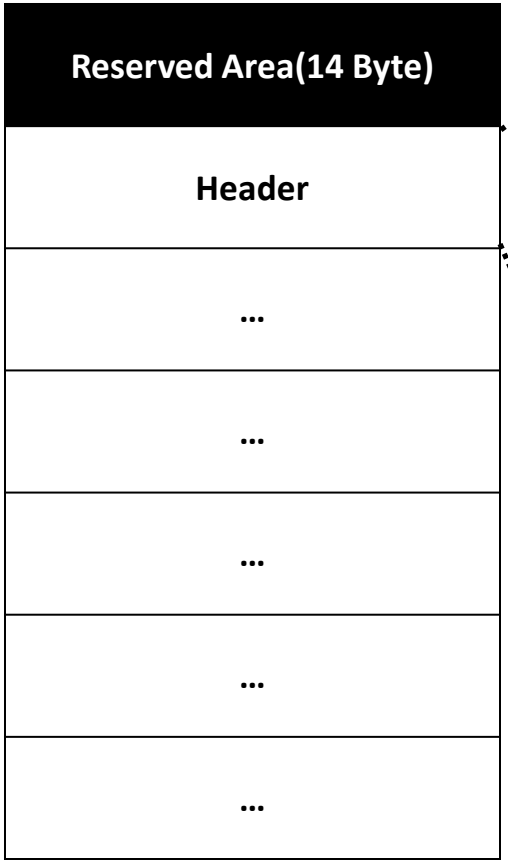
- **Find catalog header**
  - (Volume header→catalogfile→startblock) * (Volume header→blocksize) = catalog header offset
- **Volume Header**
  - Block Size = Offset 41~44





Catalog File(B-tree)

- **File Header**

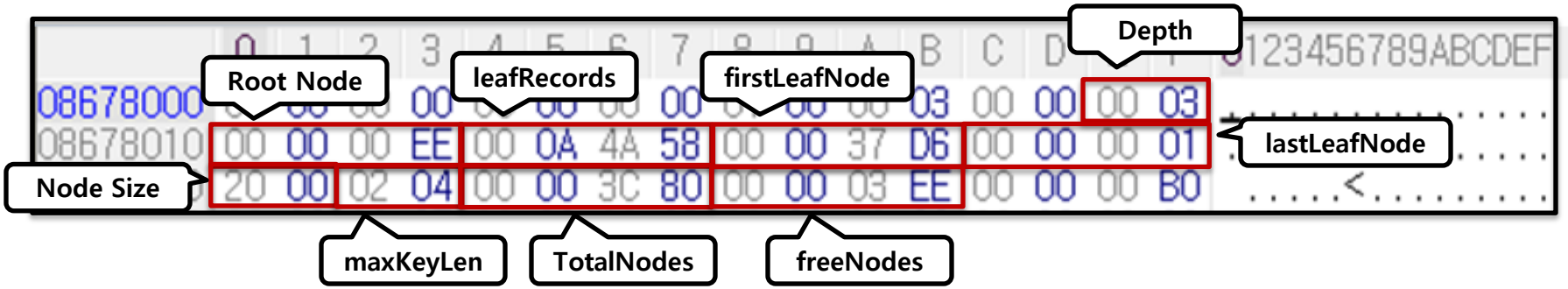



```
struct HFSPlusCatalogHeader{
    UINT8 depth[2];
    UINT8 rootNode[4];
    UINT8 leafRecords[4];
    UINT8 firstLeafNode[4];
    UINT8 lastLeafNode[4];
    UINT8 nodesize[2];
    UINT8 maxKeyLen[2];
    UINT8 totalNodes[4];
    UINT8 freeNodes[4];
    UINT8 res[2];
    UINT8 clumpSize[4];
    UINT8 type;
    UINT8 compType;
    UINT8 attr[4];
    UINT8 res2[64];
};
```

```
08678000 00 00 00 00 00 00 00 00 01 00 00 03 00 00 00 03 ................
08678010 00 00 00 EE 00 0A 4A 58 00 00 37 D6 00 00 00 01 ......JX..7.....
08678020 20 00 02 04 00 00 3C 80 00 00 03 EE 00 00 00 B0  .....<.........
08678030 00 00 00 CF 00 00 00 06 00 00 00 00 00 00 00 00 ................
```

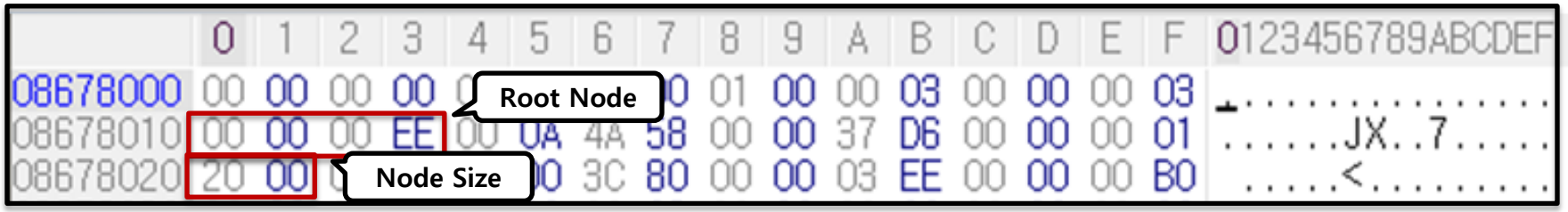
Catalog File(Find Root Node)

- **Header**



Catalog File(Find Root Node)

- **Find root node**



0xEE * 0x2000 = 0x1DC000

0x8678000 + 0x1DC000 = 0x8854000

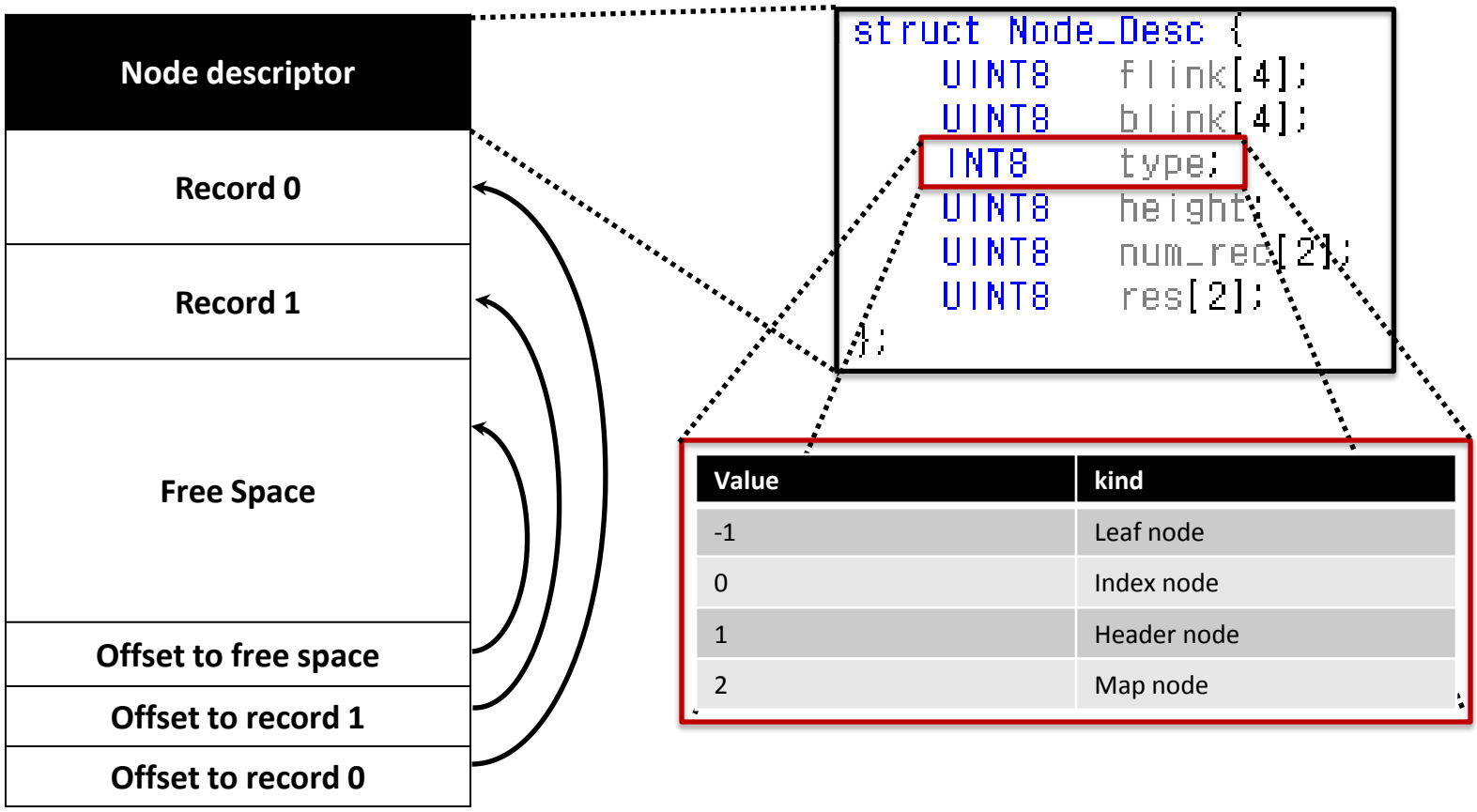
Catalog File(Find Root Node)

- **The Root Node**

```
08854000 00 00 00 00 00 00 00 00 00 03 00 4A 00 00 00 08 ...........J....
08854010 00 00 00 01 00 01 00 43 00 00 37 D4 00 36 00 00 .......C..7..6..
08854020 00 C7 00 18 00 63 00 6F 00 6D 00 2E 00 61 00 70 .....c.o.m...a.p
08854030 00 70 00 6C 00 65 00 2E 00 73 00 70 00 69 00 6E .p.l.e...s.p.i.n
08854040 00 64 00 75 00 6D 00 70 00 2E 00 70 00 6C 00 69 .d.u.m.p...p.l.i
08854050 00 73 00 74 00 00 02 02 00 26 00 00 09 30 00 10 .s.t.....&...0..
08854060 00 6B 00 65 00 79 00 65 00 64 00 6F 00 62 00 6A .k.e.y.e.d.o.b.j
08854070 00 65 00 63 00 74 00 73 00 2E 00 6E 00 69 00 62 .e.c.t.s...n.i.b
08854080 00 00 00 ED 00 06 00 00 14 8F 00 00 00 00 03 7B ...............{
08854090 00 32 00 00 1D 49 00 16 00 5F 00 78 00 67 00 72 .2...I..._.x.g.r
088540A0 00 69 00 64 00 63 00 6F 00 6E 00 74 00 72 00 6F .i.d.c.o.n.t.r.o
088540B0 00 6C 00 6C 00 65 00 72 00 2E 00 70 00 6C 00 69 .l.l.e.r...p.l.i
088540C0 00 73 00 74 00 00 01 D1 00 22 00 00 2F EC 00 0E .s.t.....".../...
088540D0 00 5F 00 43 00 6F 00 64 00 65 00 53 00 69 00 67 ._.C.o.d.e.S.i.g
088540E0 00 6E 00 61 00 74 00 75 00 72 00 65 00 00 02 D8 .n.a.t.u.r.e....
088540F0 00 06 00 00 47 10 00 00 00 00 03 BB 00 1A 00 00 ....G...........
08854100 58 20 00 0A 00 31 00 31 00 38 00 33 00 30 00 2E X ...1.1.8.3.0..
```

Catalog File(Root Node)

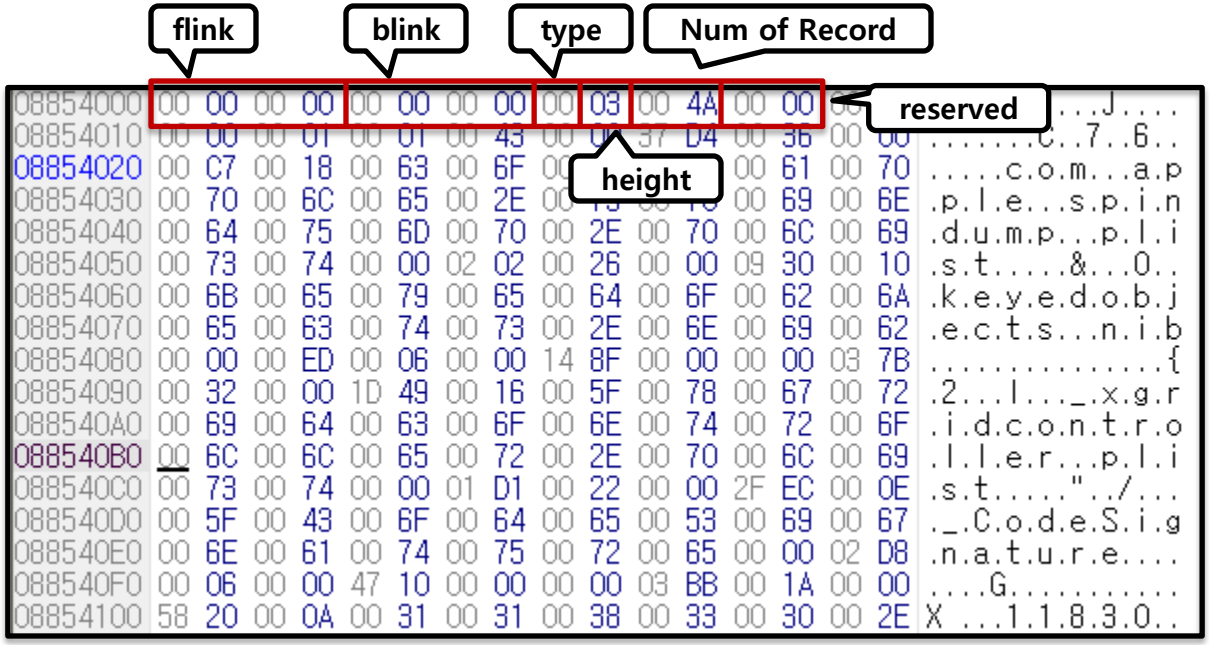
- **Node Structure (Index & Leaf node)**



Value	kind
-1	Leaf node
0	Index node
1	Header node
2	Map node

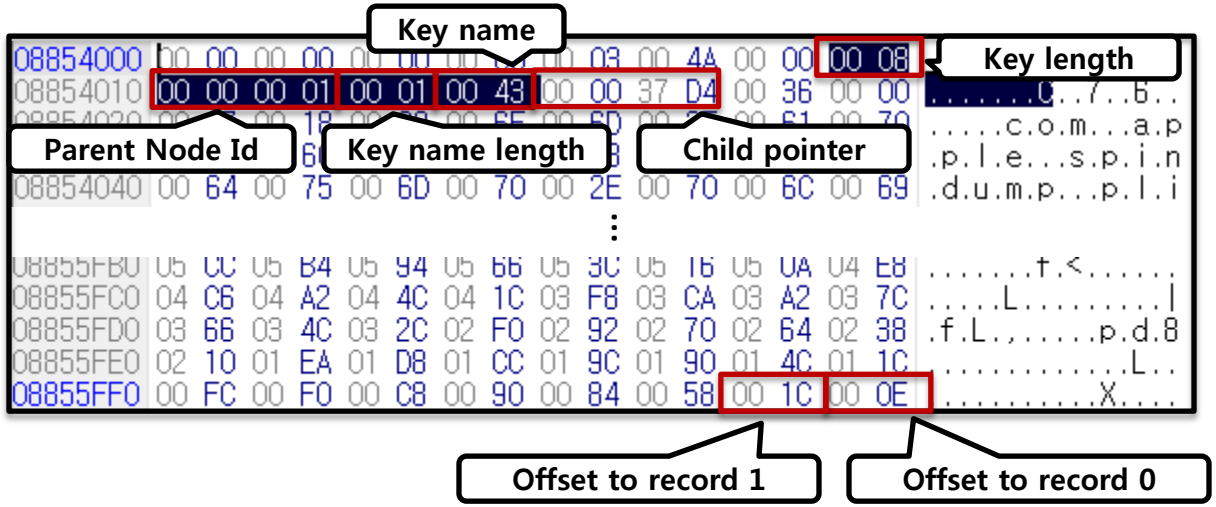
Catalog File(Root Node)

- **Node Structure sample**
  - Node Descriptor of root node



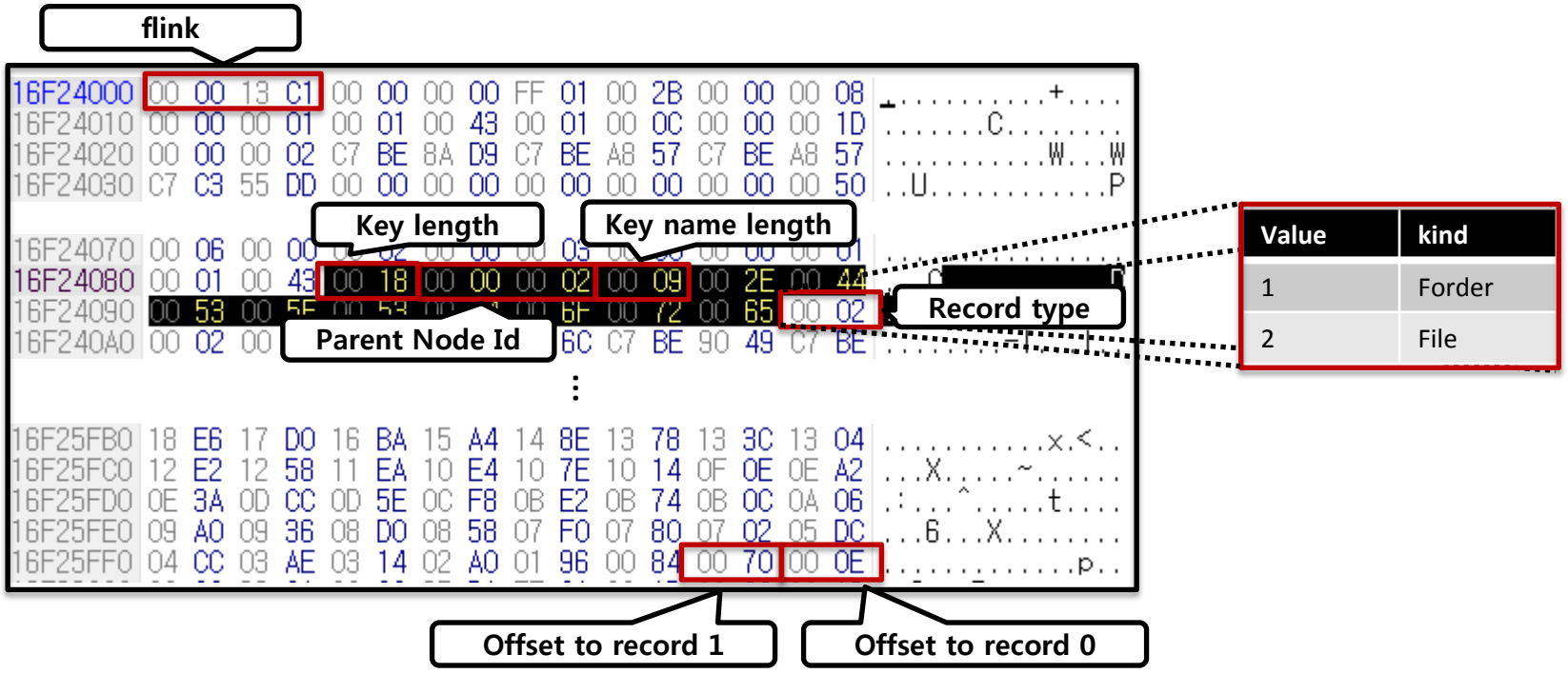
Catalog File(Root Node)

- **Index node**
  - Offset to Record'n' = node[nodesize - (n + 1) * 2];



Catalog File(Root Node)

- **Leaf node**



Catalog File (Node Traverse)

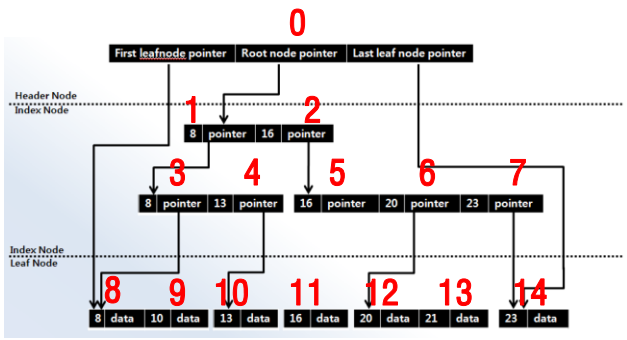
- **Find root node**
  - Index node or leaf node
  - Traverse start!

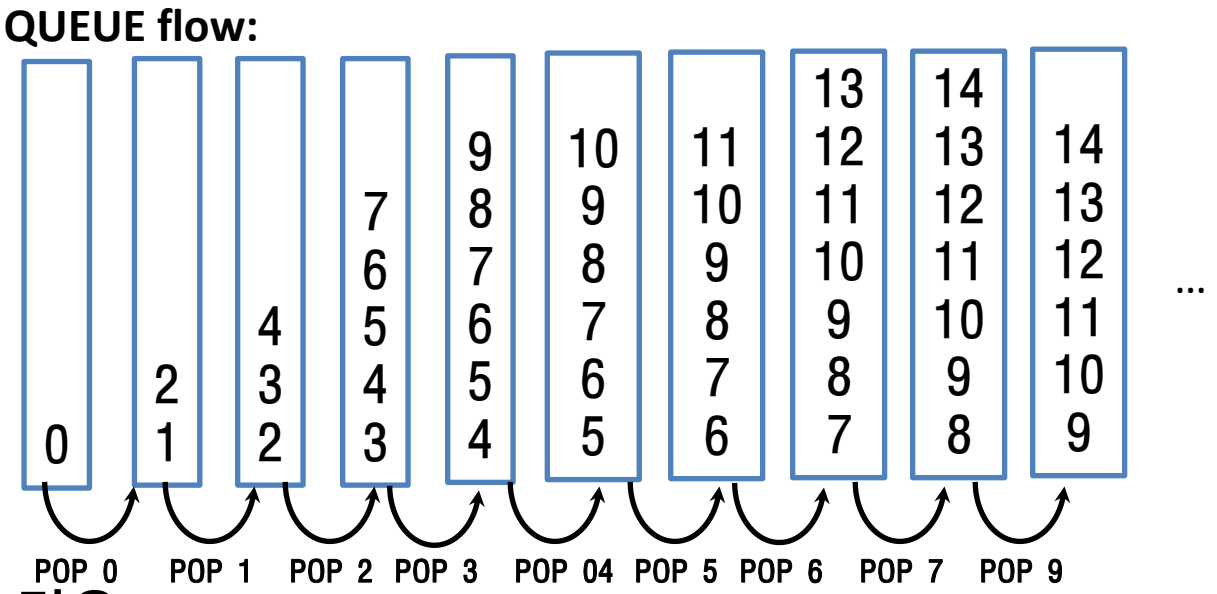
- **Index node일 경우**
  - Record 조사
  - Next node = child node(pointer)

- **Leaf node일 경우**
  - Record 조사
  - Next node = next leaf node(flink)

Catalog File (Node Traverse)

- **Node Traverse**



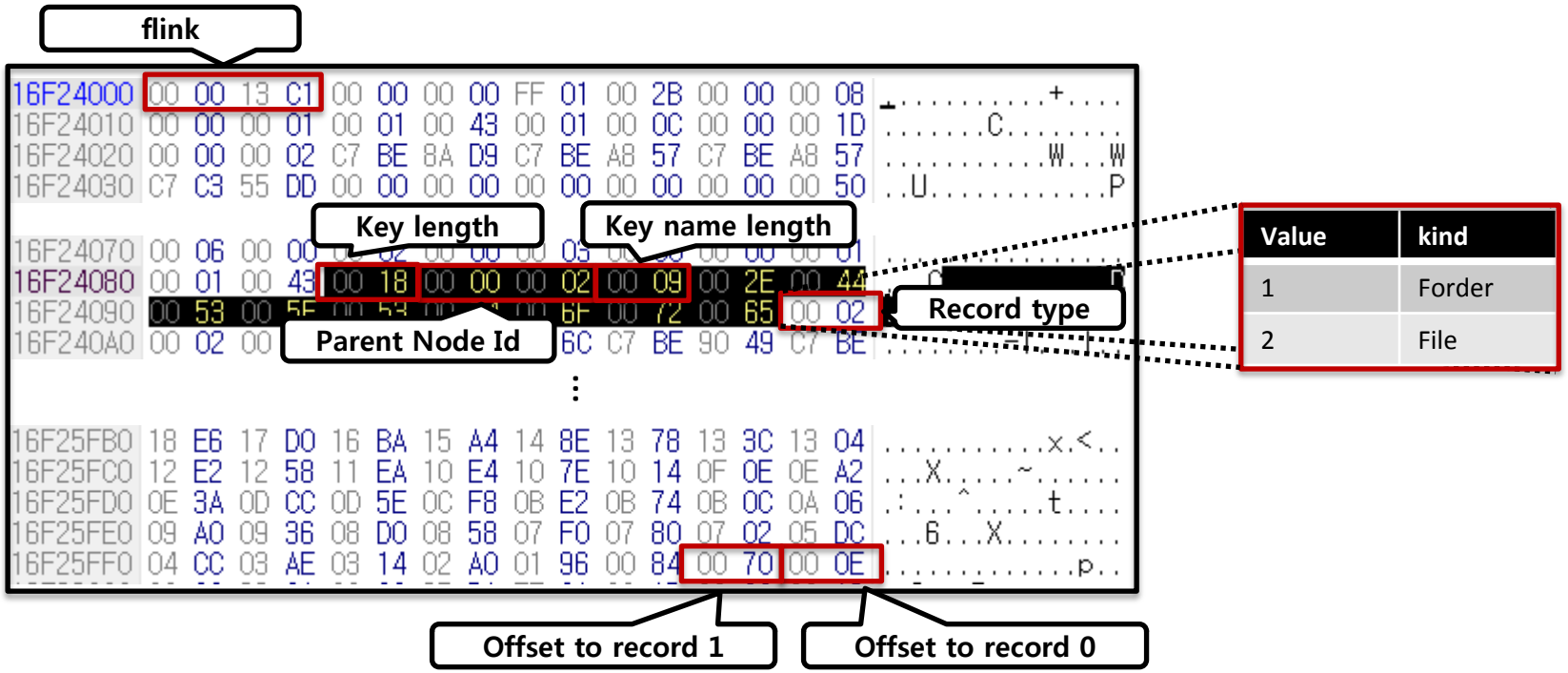


- **Index Node 가 POP 된 경우**
  - PUSH all child node
- **Leaf Node 가 POP 된 경우**
  - Read Record

Data Extract

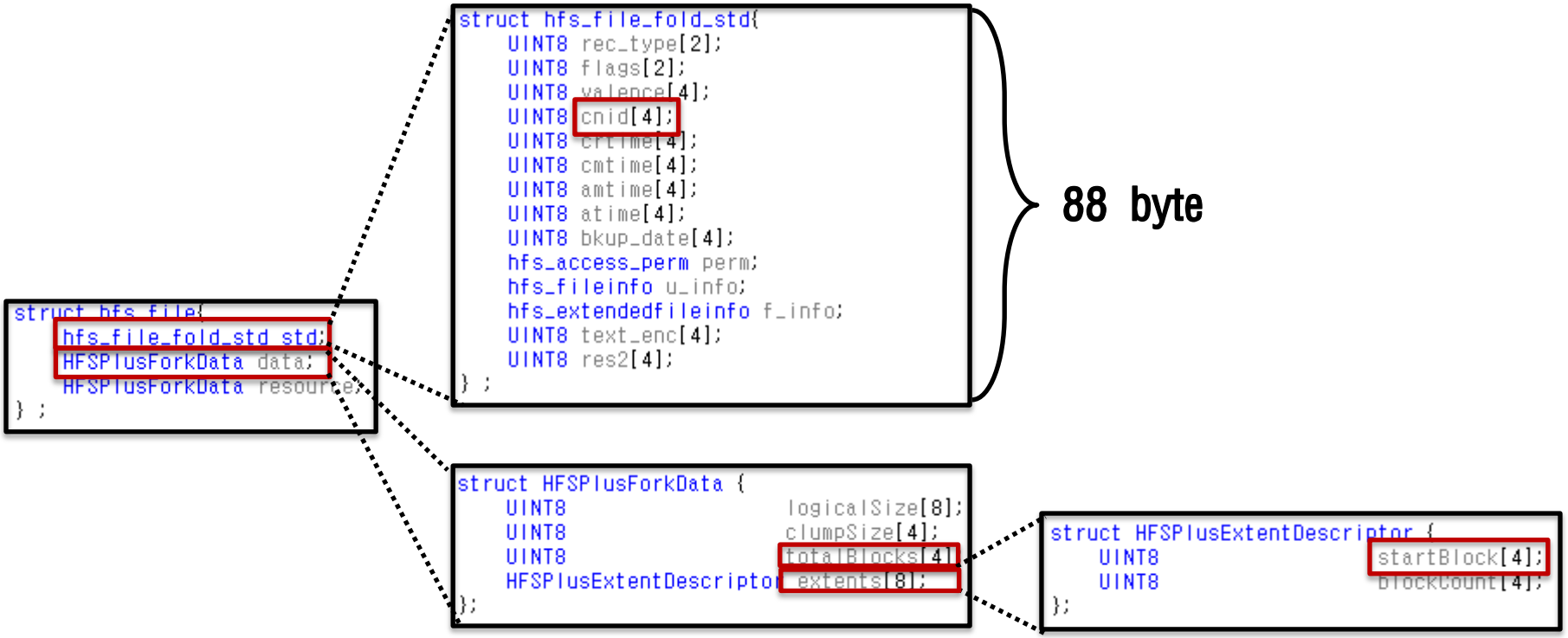
Data Extract

- 데이터 추출 (by CNID)
  - Leaf record



Data Extract

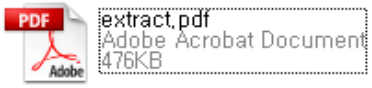
- **데이터 추출 (by CNID)**
  - Leaf record



Data Extract

- 데이터 추출
  - Sleuthkit – icat

```
C:₩sleuthkit>icat -f hfs "f:₩Mac 연동자료₩Raw₩MacRaw.dd.001"  304112 >extract.pdf
```



Deleted File

대상 데이터

- **추출 파일**
  - Cdto_2.3.zip →cnid = 341343
  - FilenoriSetup.exe → cnid = 343124
  - PurpGuy.gif → cnid = 80970
  - 스택에 관하여.pdf → cnid = 340109

결과 확인

- **Cdto_2.3.zip →cnid = 341343**

  - Record

```
016FCCFE0  00 00 00 00 00 00 00 00  00 1E 00 05 30 89 00 0C  ............0‰..
016FCCFF0  00 63 00 64 00 74 00 6F  00 5F 00 32 00 2E 00 33  .c.d.t.o._.2...3
016FCD000  00 2E 00 7A 00 69 00 70  00 02 00 02 00 00 00 00  ...z.i.p........
016FCD010  00 05 35 5F C7 BB 90 AB  C7 BB 90 B2 C7 BE 97 01  ..5_Ç»‰«Ç»‰²Ç¾—.
```

```
016FCCFE0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
016FCCFF0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
016FCD000  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
016FCD010  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  ................
```

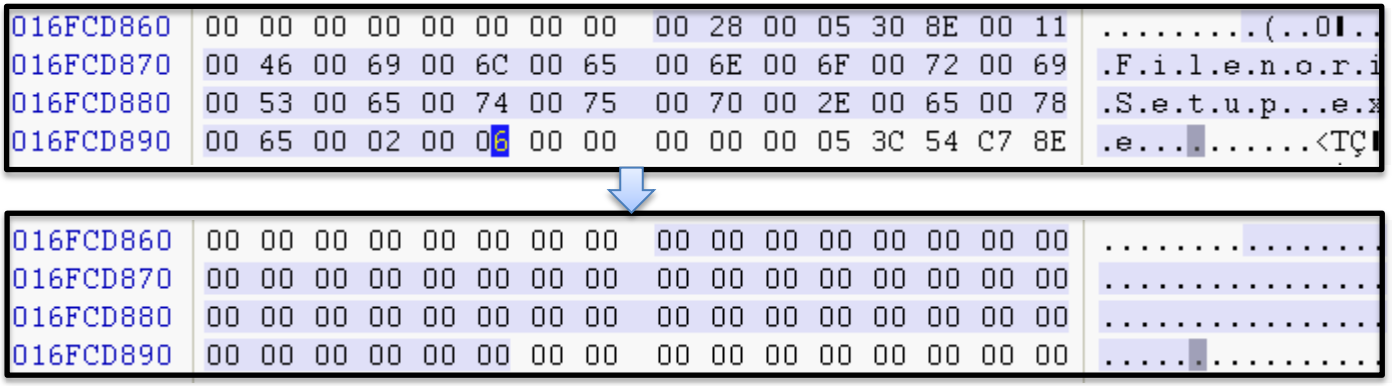
  - Data

```
1E8BA3000  50 4B 03 04 0A 00 00 00  00 00 4D 08 3B 3B 00 00  PK........M.;;..
1E8BA3010  00 00 00 00 00 00 00 00  00 00 0A 00 10 00 63 64  ..............cd
1E8BA3020  74 6F 20 64 69 73 74 2F  55 58 0C 00 42 FF BE 4A  to dist/UX..Bÿ¾J
1E8BA3030  F1 FF BE 4A F5 01 F5 01  50 4B 03 04 0A 00 00 00  ñÿ¾Jõ.õ.PK.....
```

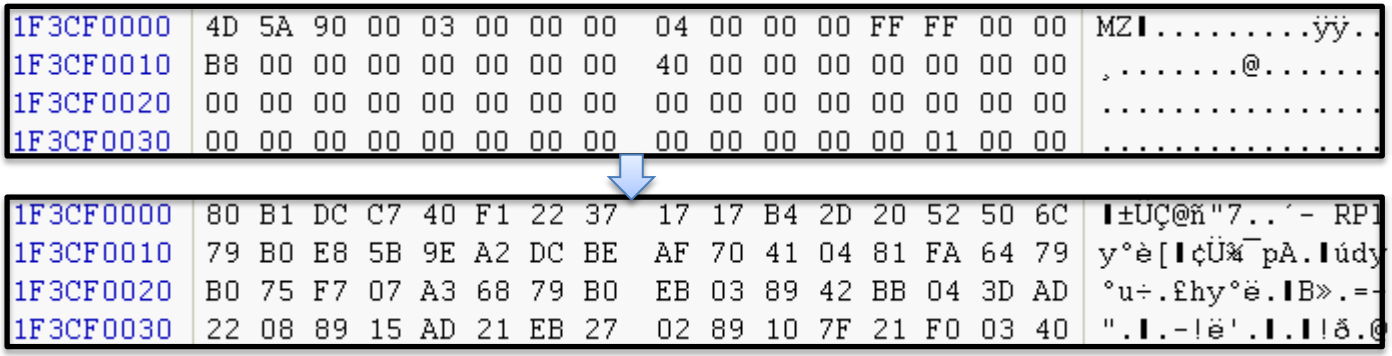
```
1E8BA3000  00 AF AD 03 00 B1 AD 03  00 B3 AD 03 00 B5 AD 03  .¯-..±-..³-..µ-
1E8BA3010  00 B7 AD 03 00 B9 AD 03  00 BB AD 03 00 BD AD 03  .·-..¹-..»-..½-
1E8BA3020  00 BF AD 03 00 C1 AD 03  00 C3 AD 03 00 C5 AD 03  .¿-..Á-..Ã-..Å-
1E8BA3030  00 C7 AD 03 00 C9 AD 03  00 CB AD 03 00 CD AD 03  .Ç-..É-..Ë-..Í-
```

결과 확인

- **FilenoriSetup.exe → cnid = 343124**
  - Record



  - Data



결과 확인

- **PurpGuy.gif → cnid = 80970**
  - Record

```
00A336000  00 00 0E 60 00 00 0E 5E  FF 01 00 33 00 00 00 1C  ...`...^ÿ..3....
00A336010  00 01 3C 46 00 0B 00 50  00 75 00 72 00 70 00 47  ..<F...P.u.r.p.G
00A336020  00 75 00 79 00 2E 00 67  00 69 00 66 00 02 00 02  .u.y...g.i.f....
00A336030  00 00 00 00 00 01 3C 4A  C6 95 8E 6E C6 95 8E 6E  ......<JÆ••nÆ••n
00A336040  C7 BE 8D 12 C7 BE 9D 6B  00 00 00 00 00 00 00 00  Ç¾•.Ç¾•k........
```

↓

```
00A336000  00 00 0E 60 00 00 0E 5E  FF 01 00 31 00 00 00 1A  ...`...^ÿ..1....
00A336010  00 01 3C 46 00 0A 00 52  00 65 00 64 00 44 00 6F  ..<F...R.e.d.D.o
00A336020  00 67 00 2E 00 67 00 69  00 66 00 02 00 02 00 00  .g...g.i.f......
00A336030  00 00 00 01 3C 4B C6 95  8E 6E C6 95 8E 6E C7 BE  ....<KÆ••nÆ••nÇ¾
00A336040  8D 12 C7 EA 5F 85 00 00  00 00 00 00 00 00 00 00  •.Çê_•..........
```

  - Data

```
142857000  47 49 46 38 39 61 30 00  30 00 F7 FF 00 3A 11 7C  GIF89a0.0.÷ÿ.:.|
142857010  FF FF FF 00 00 00 FF C2  00 E6 E6 E6 84 6B AC A1  ÿÿÿ...ÿÂ.æææ•k¬¡
142857020  A0 A3 FF CB 00 41 19 80  3A 3A 3A 46 1C 74 D2 D2  £ÿË.A.•:::F.tÒÒ
142857030  D2 CE C4 DF 8D 82 9F C5  C5 C5 39 10 7B 33 0F 6D  ÒÎÄß•••ÅÅÅ9.{3.m
142857040  9E 9E 9E C1 C1 C1 24 0B  52 AA 7D 14 64 58 78 48  •••ÁÁÁ$.Rª}.dXxH
```

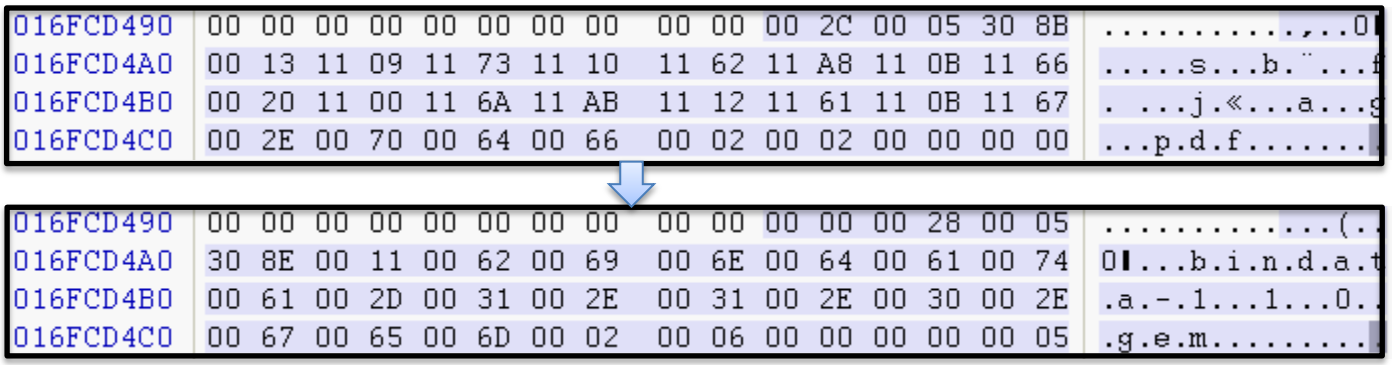
↓

```
142857000  47 49 46 38 39 61 30 00  30 00 F7 FF 00 3A 11 7C  GIF89a0.0.÷ÿ.:.|
142857010  FF FF FF 00 00 00 FF C2  00 E6 E6 E6 84 6B AC A1  ÿÿÿ...ÿÂ.æææ•k¬¡
142857020  A0 A3 FF CB 00 41 19 80  3A 3A 3A 46 1C 74 D2 D2  £ÿË.A.•:::F.tÒÒ
142857030  D2 CE C4 DF 8D 82 9F C5  C5 C5 39 10 7B 33 0F 6D  ÒÎÄß•••ÅÅÅ9.{3.m
142857040  9E 9E 9E C1 C1 C1 24 0B  52 AA 7D 14 64 58 78 48  •••ÁÁÁ$.Rª}.dXxH
```

결과 확인

- **스택에 관하여.pdf → cnid = 340109 (휴지통)**

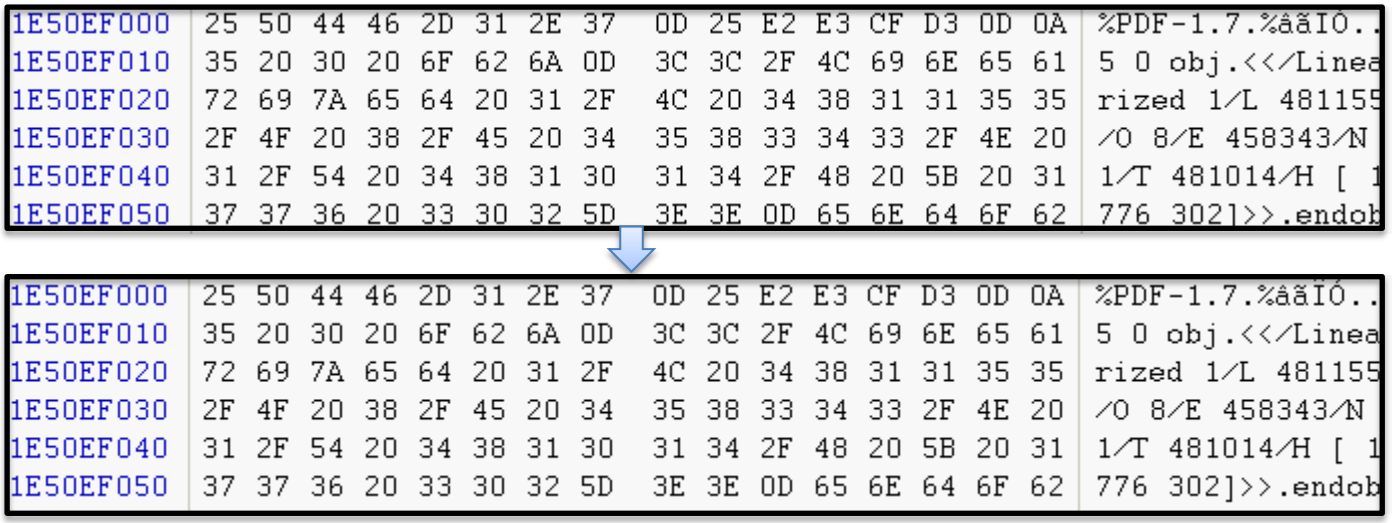
  - Record



```
016FCD490  00 00 00 00 00 00 00 00  00 00 00 2C 00 05 30 8B  ...........,..0
016FCD4A0  00 13 11 09 11 73 11 10  11 62 11 A8 11 0B 11 66  .....s...b.¨...f
016FCD4B0  00 20 11 00 11 6A 11 AB  11 12 11 61 11 0B 11 67  . ...j.«...a...g
016FCD4C0  00 2E 00 70 00 64 00 66  00 02 00 02 00 00 00 00  ...p.d.f........
```

```
016FCD490  00 00 00 00 00 00 00 00  00 00 00 00 00 28 00 05  .............(..
016FCD4A0  30 8E 00 11 00 62 00 69  00 6E 00 64 00 61 00 74  0....b.i.n.d.a.t
016FCD4B0  00 61 00 2D 00 31 00 2E  00 31 00 2E 00 30 00 2E  .a.-.1...1...0..
016FCD4C0  00 67 00 65 00 6D 00 02  00 06 00 00 00 00 00 05  .g.e.m..........
```

  - Data



```
1E50EF000  25 50 44 46 2D 31 2E 37  0D 25 E2 E3 CF D3 0D 0A  %PDF-1.7.%âãÏÓ..
1E50EF010  35 20 30 20 6F 62 6A 0D  3C 3C 2F 4C 69 6E 65 61  5 0 obj.<</Linea
1E50EF020  72 69 7A 65 64 20 31 2F  4C 20 34 38 31 31 35 35  rized 1/L 481155
1E50EF030  2F 4F 20 38 2F 45 20 34  35 38 33 34 33 2F 4E 20  /O 8/E 458343/N
1E50EF040  31 2F 54 20 34 38 31 30  31 34 2F 48 20 5B 20 31  1/T 481014/H [ 1
1E50EF050  37 37 36 20 33 30 32 5D  3E 3E 0D 65 6E 64 6F 62  776 302]>>.endob
```

```
1E50EF000  25 50 44 46 2D 31 2E 37  0D 25 E2 E3 CF D3 0D 0A  %PDF-1.7.%âãÏÓ..
1E50EF010  35 20 30 20 6F 62 6A 0D  3C 3C 2F 4C 69 6E 65 61  5 0 obj.<</Linea
1E50EF020  72 69 7A 65 64 20 31 2F  4C 20 34 38 31 31 35 35  rized 1/L 481155
1E50EF030  2F 4F 20 38 2F 45 20 34  35 38 33 34 33 2F 4E 20  /O 8/E 458343/N
1E50EF040  31 2F 54 20 34 38 31 30  31 34 2F 48 20 5B 20 31  1/T 481014/H [ 1
1E50EF050  37 37 36 20 33 30 32 5D  3E 3E 0D 65 6E 64 6F 62  776 302]>>.endob
```

결과 확인

- **파일 삭제 후 재 이미징**
  - 삭제 여부
    - 삭제 : Cdto_2.3.zip , FilenoriSetup.exe , PurpGuy.gif
    - 휴지통 : 스택에 관하여.pdf
  - 삭제 후
    - 파일 data 영역

      → 다음 파일 덮어 씌워짐
      (Cdto_2.3.zip , FilenoriSetup.exe)

      →다음 파일 데이터 남아 있음
      (PurpGuy.gif, 스택에 관하여.pdf)

    - Node record 영역

      → 전부 제거된 상태

➔ Catalog File 내의 트리 구조 재구성으로 인해 메타 데이터 영역이 남아있는 경우가 드뭄
따라서, 저널 파일을 이용한 복원 혹은 카빙을 이용한 복원 방법만을 사용 가능

질문 및 답변