# COMP 5350 / 6350
# Digital Forensics

## New Technology File System
## Locating User Generated Data

# New Technology File System Introduction

# NTFS

- New Technology File System is a complex file system that provides greater functionality, reliability, and security, and large storage device support when compared with File Allocation Table (FAT) versions
- The operating systems designed for use with NTFS include:
  - Windows NT 3.1 – 4.0
  - Windows 95
  - Windows 98
  - Windows NT
  - Windows 2000
  - Windows XP
  - Windows 7
  - Windows 8
  - Windows 10

| File System | Max File Size | Max Partition Size | File Permissions |
|---|---|---|---|
| FAT12 | 16 MiB | 16 MiB | |
| FAT16 | 2 GiB | 2 GiB | |
| FAT32 | 4 GiB | 8 TiB | |
| NTFS | 2 TiB | 256 TiB | X |
| ext2 | 2 TiB | 32 TiB | X |
| ext3 | 2 TiB | 32 TiB | X |
| ext4 | 16 TiB | 1 EiB | X |
| HFS+ | 2 GiB | 2 TiB | X |
| XFS | 8 EiB | 8 EiB | X |

# FAT vs. NTFS

- Key differences between FAT and NTFS:

| | FAT16 | FAT32 | NTFS |
|---|---|---|---|
| Volume Size (Max) | 2 GB - 4 GB | 32 GB - 2 TB | 2 TB |
| File Size (Max) | 2 GB | 4 GB | 16 EB |
| File Name Length (Max) | 8.3 / < 255 chars | < 255 chars | < 255 |
| Encoding | System | System | Unicode |
| Compression | No | No | Yes |
| Encryption | No | No | Yes |
| File Permissions | No | No | Yes |
| Built-In Security | No | No | Yes |
| Fault Tolerance | No | Minimal | Yes |

# NTFS Version History

- Version 1.0
  - ✓ Windows NT 3.1, 1993
- Version 1.1
  - ✓ Windows NT 3.51, 1995
  - ✓ File Compression
  - ✓ Named Streams
  - ✓ Access Control Lists
- Version 1.2
  - ✓ Windows NT 4.0, 1996
  - ✓ Security Descriptors
- Version 3.0
  - ✓ Windows 2000, 2000
  - ✓ Disk Quotas
  - ✓ Encrypting File System
  - ✓ Sparse Files
  - ✓ Reparse Points
  - ✓ Update Sequence Number (USN) Journaling
  - ✓ Reorganized Security Descriptors
- Version 3.1
  - ✓ Expanded the Master File Table (MFT) Entries

# NTFS Capabilities

- Self-healing
  - ✓ Detection and correction of volume corruption without running disk repair utility
- Access Control Lists (ACL)
  - ✓ Administrator set system permissions with ACLs to preemptively identify resource access
- Encrypting File System (EFS)
  - ✓ NTFS utilized DESX to provide file encryption
- Disk Quotas
  - ✓ Tracking user disk utilization and restricting disk space
- Fault Tolerance
  - ✓ Automatic disk recovery using transaction logs and journal files
- File Compression
  - ✓ Use of Lempel-Ziv (LZ) compression algorithm to compress large files for efficient disk utilization

# Additional NTFS Capabilities

- Alternate Data Streams (ADS)
  - ✓ Areas where additional, non-critical, information about files can be stored
- Security Descriptors
  - ✓ Attributes that define all security definitions of a file or directory
- Sparse Files
  - ✓ Disk allocation based on files that contain nonzero data
- Reparse Points
  - ✓ Path linking, similar to symbolic links in Linux
- Update Sequence Number (USN) Journaling
  - ✓ Recording of all volume changes

# Find NTFS Partition(s)

- For this session we will be using disk image "disk2.dd"

1 Sector = 512 bytes

```
Forensics $ sudo fdisk -l disk2.dd
Disk disk2.dd: 1.8 GiB, 1941962752 bytes, 3792896 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0xc3072e18

Device     Boot Start     End Sectors Size Id Type
disk2.dd1        2048 1026047 1024000 500M 86 NTFS volume set
```

1,024,000 Sectors * 512 Bytes = 500 MB

2,048 Sectors

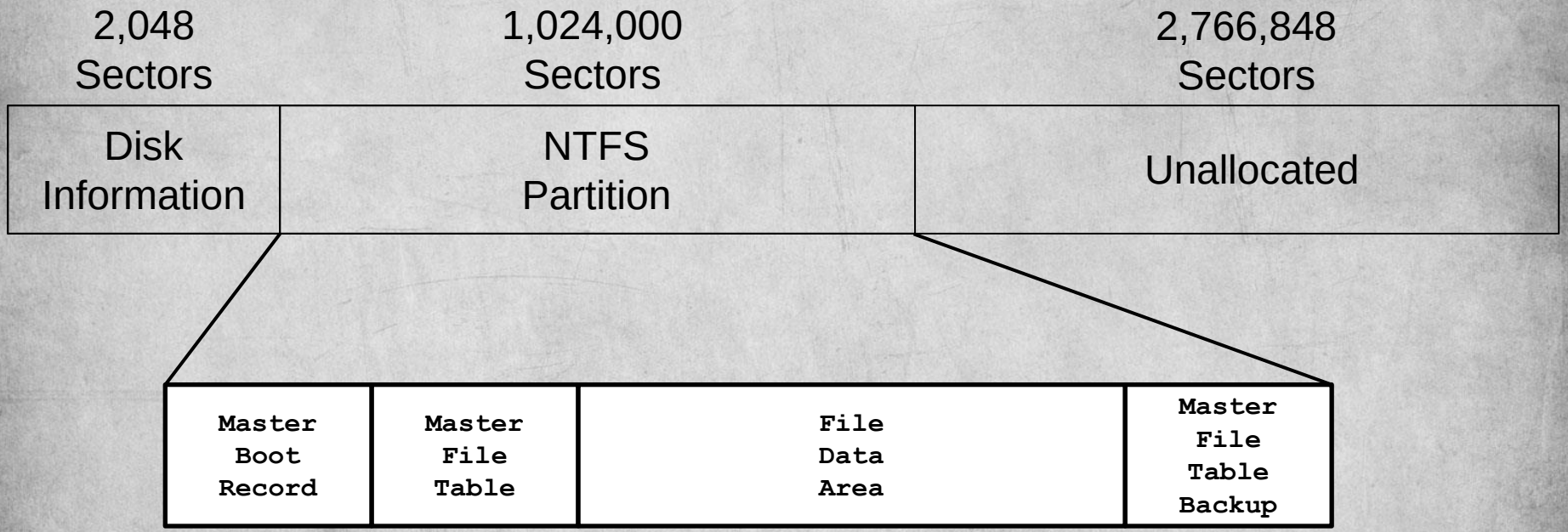
1,024,000 Sectors

2,766,848 Sectors

| Disk Information | NTFS Partition | Unallocated |
|---|---|---|

3,792,896 Sectors

# NTFS Partition Architecture



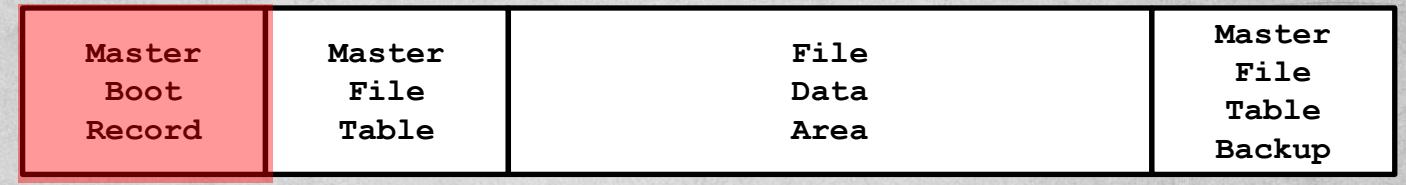

Note: The actual NTFS layout has never been published so this view is general and should not be taken as the data structure!

# NTFS Master Boot Record

# NTFS Master Boot Record

- The Master Boot Record (MBR) is the first sector of an NTFS partition
- The MBR contains a partition table and a small amount of executable code like the FAT boot sector
- The MBR finds the starting point of the partition and loads a copy the boot sector into memory

# NTFS Master Boot Record




```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

# NTFS MBR Usage

- In comparison to FAT partitions, there are many unused bytes within the NTFS MBR

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

# Master Boot Record - Bootstrap

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0xEB5290: Jump Short 52 NOP

- If bootable, jump 82 bytes to the start of boot code

# Master Boot Record – OEM ID

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0x4E54465320202020: "NTFS    "

# Master Boot Record – # Bytes / Sector

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0x200: 512 Bytes / Sector

# Master Boot Record – # Sectors / Cluster

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0x8: 8 Sectors / Cluster

$$512\frac{bytes}{sector} * 8\frac{sectors}{cluster} = 4096\frac{bytes}{cluster}$$

# Master Boot Record – Reserved Sectors

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0x0: 0 Reserved Sectors

# Master Boot Record – Set to 0

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0x0: 0

# Master Boot Record – Not Used By NTFS

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0x0: 0

# Master Boot Record – Media Descriptor

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0xF8: Fixed Disk – Hard Disk Partition

# Master Boot Record – Always 0

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0x0: 0

# Master Boot Record – # Sectors / Track

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0x3F: 63 Sectors / Track

# Master Boot Record – # Drive Heads

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0xFF: 255 Drive Heads

# Master Boot Record – # Sector Before Partition

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0x800: 2048 Sectors Before NTFS Partition Starts

# Master Boot Record – Not Used By NTFS

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0x0: Not Used by NTFS

# Master Boot Record – Not Used By NTFS

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0x80: Not Used by NTFS

# Master Boot Record – Not Used By NTFS

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0x31FFF: 204,799 Sectors

$$204799\,sectors * 512\frac{bytes}{sector} = 100MB$$

# Master Boot Record – $MFT Start Cluster*

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

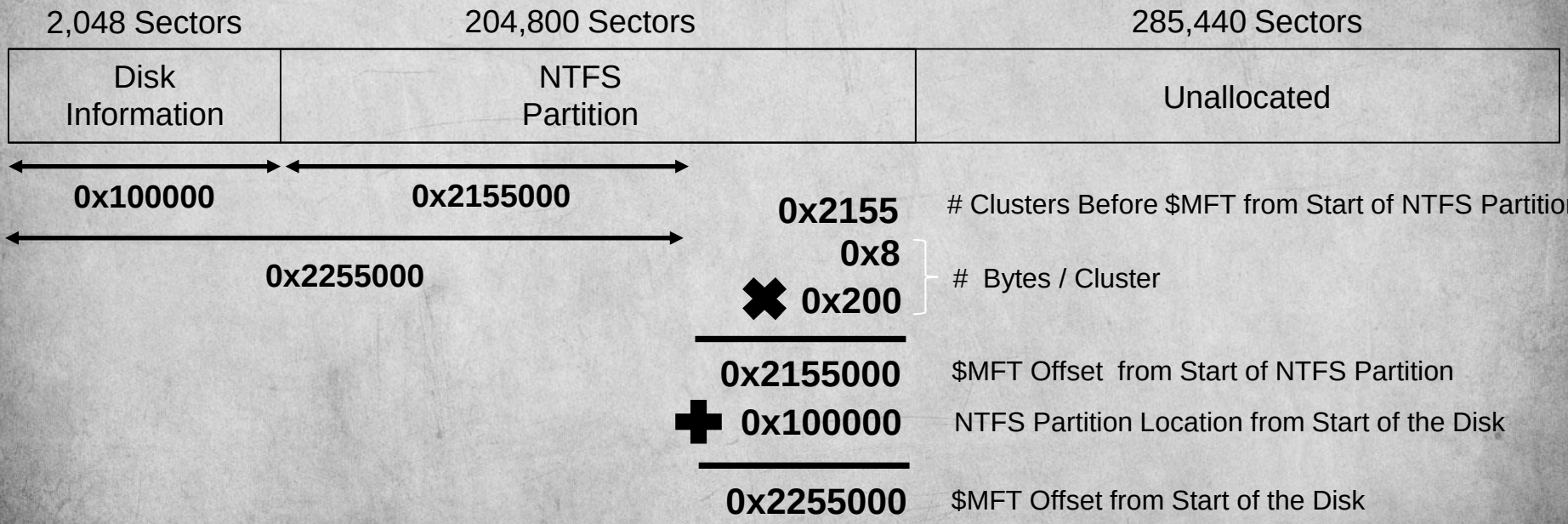
```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0x2155: $MFT Starts at Cluster 0x2155

# Master Boot Record – $MFT Start Cluster

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
```

0x2155: $MFT Starts at Cluster 0x2155



# Master Boot Record – $MFT Backup Start

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

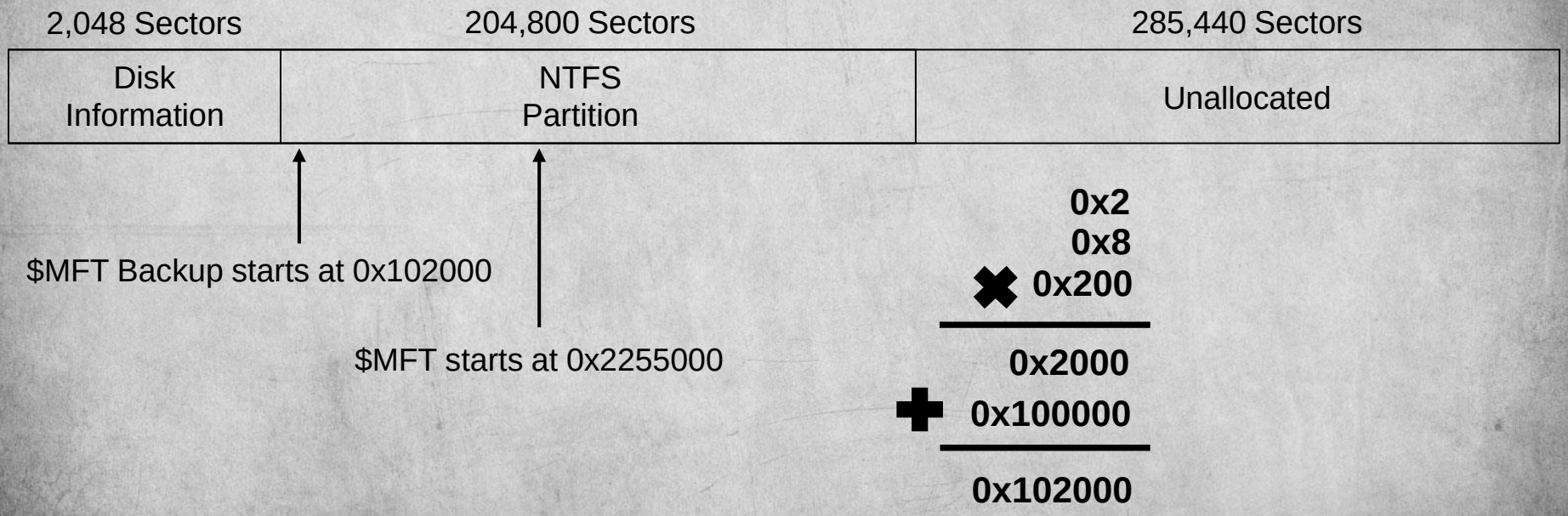
```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0x2: $MFT Backup Starts at Cluster 2

# Master Boot Record – $MFT Backup Start

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
```

0x2155: $MFT Starts at Cluster 0x2155



# Master Boot Record – # Cluster / File Record

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0xF6: 246 Clusters / $MFT Record

# Master Boot Record – # Cluster / Index Buffer

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0x1: 1 Cluster / Index Buffer

# Master Boot Record – Not Used By NTFS

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0x0: Not Used By NTFS

# Master Boot Record – Volume Serial Number

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

0xD2210238300238C4: D2210238-300238C4

# Master Boot Record – Checksum

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

NTFS Checksum - Not Implemented

# Master Boot Record – Boot Code

| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    ....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|

001001e0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
001001f0  00 00 00 00 00 00 8a 01  a7 01 bf 01 00 00 55 aa  |..............U.|
```

- If the partition is bootable, then this code will run

# Master Boot Record – Boot Sector Signature

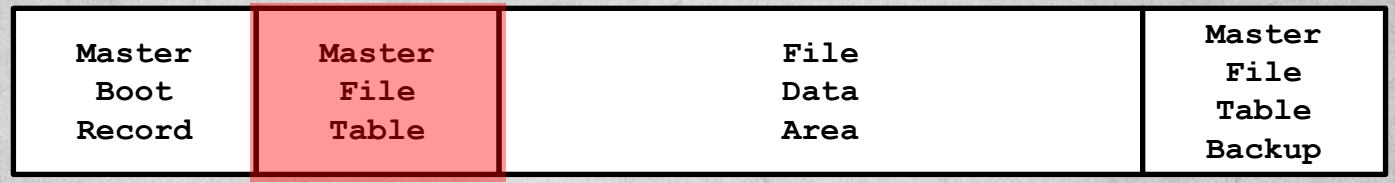
| NTFS Master Boot Record | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Bootstrap Jump Command* | 0000h | 3 |
| OEM Identification* | 0003h | 8 |
| # Bytes / Sector | 000Bh | 2 |
| # Sectors / Cluster | 000Dh | 1 |
| # Reserved Sectors | 000Eh | 2 |
| Always 0 | 0010h | 3 |
| Not Used By NTFS | 0013h | 2 |
| Media Descriptor | 0015h | 1 |
| Always 0 | 0016h | 2 |
| # Sectors / Track | 0018h | 2 |
| # Drive Heads | 001Ah | 2 |
| # Sectors Before Partition | 001Ch | 4 |
| Not Used By NTFS | 0020h | 4 |
| Not Used By NTFS | 0024h | 4 |
| # Sectors | 0028h | 8 |
| $MFT Cluster Number | 0030h | 8 |
| $MFTMirr Cluster Number | 0038h | 8 |
| # Clusters / File Record | 0040h | 4 |
| # Clusters / Index Buffer | 0044h | 1 |
| Not Used By NTFS | 0045h | 3 |
| Volume Serial Number | 0048h | 8 |
| Checksum | 0050h | 4 |
| Bootstrap Code* | 003Eh | 426 |
| Boot Sector Signature | 01FEh | 2 |
| * Big Endian | | |

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    ....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
00100040  f6 00 00 00 01 00 00 00  d2 21 02 38 30 02 38 c4  |.........!.80.8.|
00100050  00 00 00 00 fa 33 c0 8e  d0 bc 00 7c fb 68 c0 07  |.....3.....|.h..|
00100060  1f 1e 68 66 00 cb 88 16  0e 00 66 81 3e 03 00 4e  |..hf......f.>..N|
00100070  54 46 53 75 15 b4 41 bb  aa 55 cd 13 72 0c 81 fb  |TFSu..A..U..r...|
00100080  55 aa 75 06 f7 c1 01 00  75 03 e9 dd 00 1e 83 ec  |U.u.....u.......|
00100090  18 68 1a 00 b4 48 8a 16  0e 00 8b f4 16 1f cd 13  |.h...H..........|
001000a0  9f 83 c4 18 9e 58 1f 72  e1 3b 06 0b 00 75 db a3  |.....X.r.;...u..|
001000b0  0f 00 c1 2e 0f 00 04 1e  5a 33 db b9 00 20 2b c8  |........Z3... +.|
001000c0  66 ff 06 11 00 03 16 0f  00 8e c2 ff 06 16 00 e8  |f...............|
001000d0  4b 00 2b c8 77 ef b8 00  bb cd 1a 66 23 c0 75 2d  |K.+.w......f#.u-|
001000e0  66 81 fb 54 43 50 41 75  24 81 f9 02 01 72 1e 16  |f..TCPAu$....r..|
001000f0  68 07 bb 16 68 52 11 16  68 09 00 66 53 66 53 66  |h...hR..h..fSfSf|
```

```
001001e0  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
001001f0  00 00 00 00 00 00 8a 01  a7 01 bf 01 00 00 55 aa  |..............U.|
```

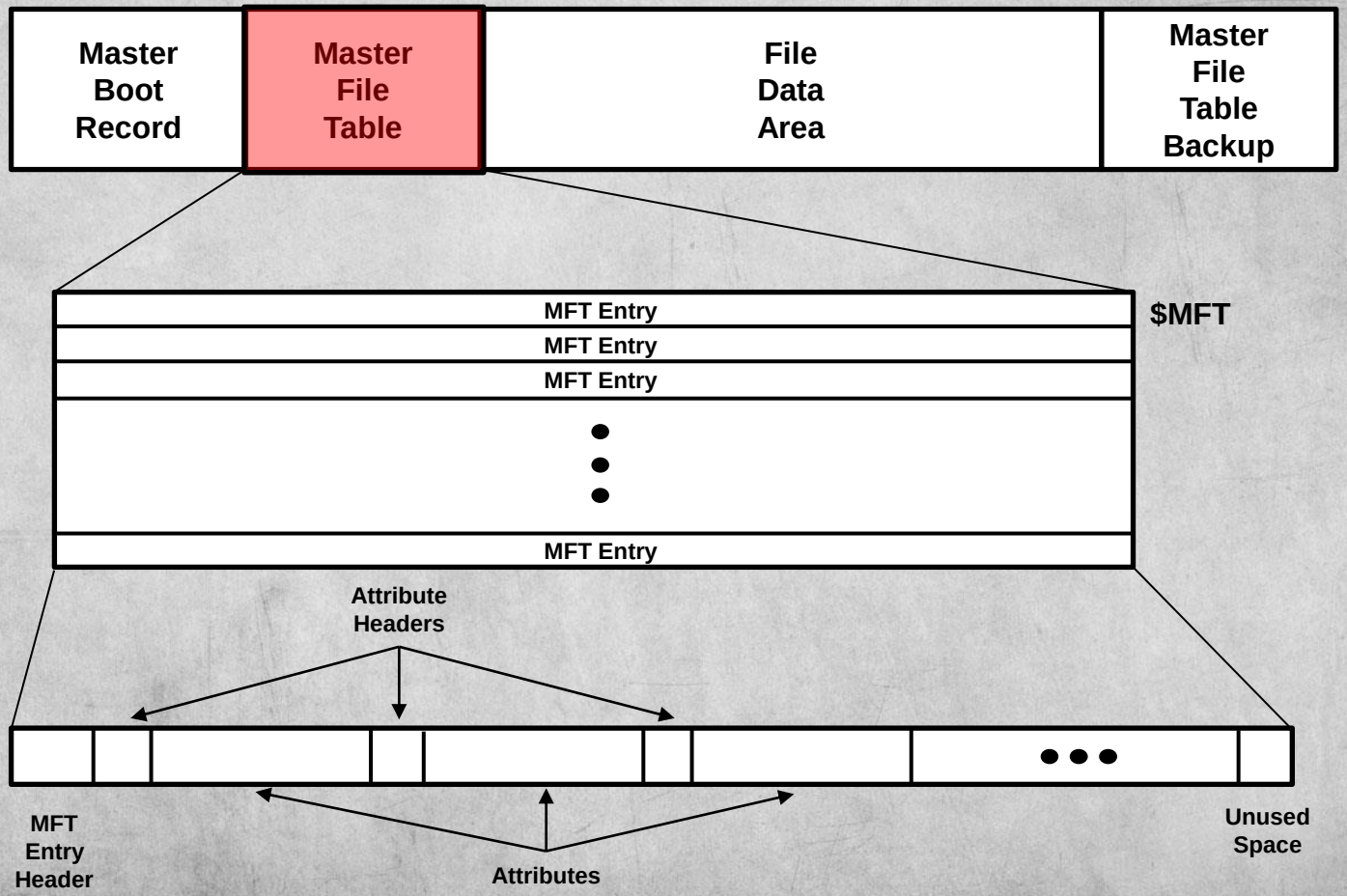
# NTFS Master File Table

# Master File Table




- Unlike legacy file systems like FAT, which only have basic functionality, NTFS defines file system objects with attributes such as:
  - ✓ Compression
  - ✓ Encryption
  - ✓ Security
- A Master File Table is composed of 1024-byte entries for every data object in the NTFS file system including files, folders, and applications and can be augmented with additional entries if necessary
- The MFT initially accounts for 1/8 of the overall size of the NTFS partition and can grow to 1/2

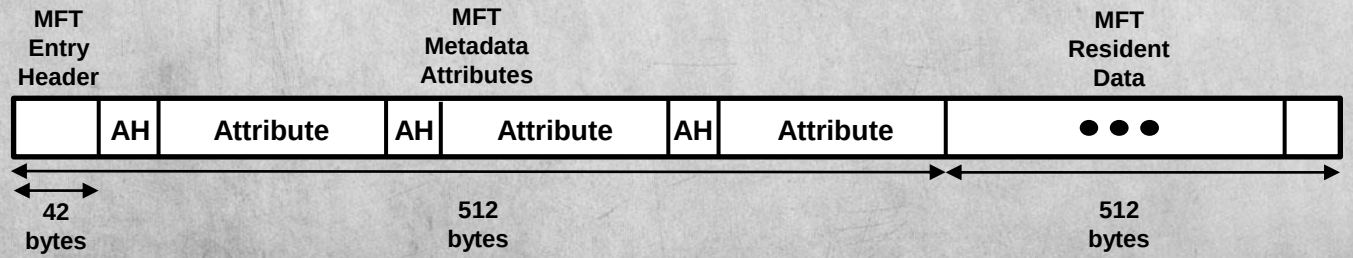
# MFT Architecture



# Master File Table Entry

| NTFS MFT Header | | |
|---|---|---|
| Description | Offset | Bytes |
| FILE Signature | 0000h | 4 |
| Fix-Up Array Offset | 0004h | 2 |
| Fix-Up Array Entries | 0006h | 2 |
| Logfile Sequence Number | 0008h | 8 |
| Incremental Sequence Value | 0010h | 2 |
| Hard Link Count | 0012h | 2 |
| Attribute Start Offset | 0014h | 2 |
| In Use Flags | 0016h | 2 |
| MFT Entry Used Size | 0018h | 4 |
| MFT Entry Allocated Size | 001Ch | 4 |
| Base Record File Reference | 0020h | 8 |
| Next Attribute ID | 0028h | 2 |
| Fix-Up Code and Attribute | 002Ah | 2 |
| $MFT File Record Number | 002Ch | 4 |
| Fix-Up Code and Attributes | | 982 |

- The MFT is 1024 bytes long with the first 42 bytes making up the MFT header
- The first 512 bytes of an MFT entry contains file metadata attributes
- For files that are 512 bytes and less, the second 512 bytes can be used to store a file in the MFT entry
- The first 39 MFT entries are reserved for system files



# MFT Entry Definitions

- **Fix-Up Array**
  - ✓ **NTFS fix-up arrays detect file system errors with MFT sequence numbers**
- **Logfile Sequence Number**
  - ✓ **A tracking number that changes every time a record is modified**
- **Sequence Number**
  - ✓ **Number of times the MFT record has been reused**
- **Hard Link Count**
  - ✓ **Counts the number of directory entries for the current record**

| NTFS MFT Header | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| FILE Signature | 0000h | 4 |
| Fix-Up Array Offset | 0004h | 2 |
| Fix-Up Array Entries | 0006h | 2 |
| Logfile Sequence Number | 0008h | 8 |
| Incremental Sequence Value | 0010h | 2 |
| Hard Link Count | 0012h | 2 |
| Attribute Start Offset | 0014h | 2 |
| In Use Flags | 0016h | 2 |
| MFT Entry Used Size | 0018h | 4 |
| MFT Entry Allocated Size | 001Ch | 4 |
| Base Record File Reference | 0020h | 8 |
| Next Attribute ID | 0028h | 2 |
| Fix-Up Code and Attribute | 002Ah | 2 |
| $MFT File Record Number | 002Ch | 4 |
| Fix-Up Code and Attributes | | 982 |

# Master File Table Entry Definitions

| NTFS MFT Header | | |
|---|---|---|
| Description | Offset | Bytes |
| FILE Signature | 0000h | 4 |
| Fix-Up Array Offset | 0004h | 2 |
| Fix-Up Array Entries | 0006h | 2 |
| Logfile Sequence Number | 0008h | 8 |
| Incremental Sequence Value | 0010h | 2 |
| Hard Link Count | 0012h | 2 |
| Attribute Start Offset | 0014h | 2 |
| In Use Flags | 0016h | 2 |
| MFT Entry Used Size | 0018h | 4 |
| MFT Entry Allocated Size | 001Ch | 4 |
| Base Record File Reference | 0020h | 8 |
| Next Attribute ID | 0028h | 2 |
| Fix-Up Code and Attribute | 002Ah | 2 |
| $MFT File Record Number | 002Ch | 4 |
| Fix-Up Code and Attributes | | 982 |

- **Attribute Start Offset**
  - ✓ Number of bytes before an attribute
- **Flags**
  - ✓ MFT header flags specifying file status
    - 0x01 – Record In Use
    - 0x02 – Directory
    - 0x04 – Don’t Know
    - 0x08 – Don’t Know
- **Allocated Size**
  - ✓ Space taken up by a record
  - ✓ Usually a multiple of cluster size
- **Base Record File Reference**
  - ✓ Zero for base MFT records
  - ✓ Non-zero when an MFT reference

# $MFT Entry

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((2048*512)) -n $((1*512))
00100000  eb 52 90 4e 54 46 53 20  20 20 20 00 02 08 00 00  |.R.NTFS    .....|
00100010  00 00 00 00 00 f8 00 00  3f 00 ff 00 00 08 00 00  |........?.......|
00100020  00 00 00 00 80 00 00 00  ff 1f 03 00 00 00 00 00  |................|
00100030  55 21 00 00 00 00 00 00  02 00 00 00 00 00 00 00  |U!..............|
```

2,048 Sectors | 204,800 Sectors | 285,440 Sectors

| Disk Information | NTFS Partition | Unallocated |
|---|---|---|

$MFT starts at 0x2255000

$MFT Backup starts at 0x102000

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((70312*512)) -n $((1*512))
02255000  46 49 4c 45 30 00 03 00  63 15 10 00 00 00 00 00  |FILE0...c.......|
02255010  01 00 01 00 38 00 01 00  a0 01 00 00 00 04 00 00  |....8...........|
02255020  00 00 00 00 00 00 00 00  07 00 00 00 00 00 00 00  |................|
02255030  02 00 00 00 00 00 00 00  10 00 00 00 60 00 00 00  |............`...|
02255040  00 00 18 00 00 00 00 00  48 00 00 00 18 00 00 00  |........H.......|
02255050  79 93 25 01 2c 87 d6 01  79 93 25 01 2c 87 d6 01  |y.%.,...y.%.,...|
*
02255070  06 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
02255080  00 00 00 00 00 00 01 00  00 00 00 00 00 00 00 00  |................|
02255090  00 00 00 00 00 00 00 00  30 00 00 00 68 00 00 00  |........0...h...|
022550a0  00 00 18 00 00 00 03 00  4a 00 00 00 18 00 01 00  |........J.......|
022550b0  05 00 00 00 00 00 05 00  79 93 25 01 2c 87 d6 01  |........y.%.,...|
022550c0  79 93 25 01 2c 87 d6 01  79 93 25 01 2c 87 d6 01  |y.%.,...y.%.,...|
022550d0  79 93 25 01 2c 87 d6 01  00 40 00 00 00 00 00 00  |y.%.,....@......|
022550e0  00 40 00 00 00 00 00 00  06 00 00 00 00 00 00 00  |.@..............|
022550f0  04 03 24 00 4d 00 46 00  54 00 00 00 00 00 00 00  |..$.M.F.T.......|
02255100  80 00 00 00 48 00 00 00  01 00 40 00 00 00 06 00  |....H.....@.....|
02255110  00 00 00 00 00 00 00 00  3f 00 00 00 00 00 00 00  |........?.......|
02255120  40 00 00 00 00 00 00 00  00 00 04 00 00 00 00 00  |@...............|
02255130  00 00 04 00 00 00 00 00  00 00 04 00 00 00 00 00  |................|
02255140  21 40 55 21 00 00 00 00  b0 00 00 00 50 00 00 00  |!@U!........P...|
02255150  01 00 40 00 00 00 05 00  00 00 00 00 00 00 00 00  |..@.............|
02255160  01 00 00 00 00 00 00 00  40 00 00 00 00 00 00 00  |........@.......|
02255170  00 20 00 00 00 00 00 00  08 10 00 00 00 00 00 00  |. ..............|
02255180  08 10 00 00 00 00 00 00  21 01 54 21 21 01 d1 de  |........!.T!!...|
02255190  00 00 00 00 00 00 00 00  ff ff ff ff 00 00 00 00  |................|
022551a0  00 00 04 00 00 00 00 00  00 00 04 00 00 00 00 00  |................|
022551b0  00 00 04 00 00 00 00 00  21 40 55 21 00 00 00 00  |........!@U!....|
022551c0  b0 00 00 00 50 00 00 00  01 00 40 00 00 00 05 00  |....P.....@.....|
022551d0  00 00 00 00 00 00 00 00  01 00 00 00 00 00 00 00  |................|
022551e0  40 00 00 00 00 00 00 00  00 20 00 00 00 00 00 00  |@........ ......|
022551f0  08 10 00 00 00 00 00 00  08 10 00 00 00 00 02 00  |................|
```

# NTFS Standard File Attributes

# $MFT Header Details

- Offset to first attribute
  - ✓ 0x0038 => 56 bytes
- Flags
  - ✓ 0x0001 => Record in Use
- MFT Record Size – Actual
  - ✓ 0x1A0 => 416 bytes
- MFT Record Size – Allocated
  - ✓ 0x400 => 1024 bytes
- Next Attribute ID
  - ✓ 0x7 => $Boot
- MFT Record Number
  - ✓ 0x0000 => 0

```
Forensics $ sudo dd if=disk2.dd bs=512 | hexdump -C -s $((70312*512)) -n $((1*512))
02255000  46 49 4c 45 30 00 03 00  63 15 10 00 00 00 00 00  |FILE0...c.......|
02255010  01 00 01 00 38 00 01 00  a0 01 00 00 00 04 00 00  |....8...........|
02255020  00 00 00 00 00 00 00 00  07 00 00 00 00 00 00 00  |................|
02255030  02 00 00 00 00 00 00 00  10 00 00 00 60 00 00 00  |............`...|
02255040  00 00 18 00 00 00 00 00  48 00 00 00 18 00 00 00  |........H.......|
02255050  79 93 25 01 2c 87 d6 01  79 93 25 01 2c 87 d6 01  |y.%.,...y.%.,...|
*
02255070  06 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
02255080  00 00 00 00 00 01 00 00  00 00 00 00 00 00 00 00  |................|
02255090  00 00 00 00 00 00 00 00  30 00 00 00 68 00 00 00  |........0...h...|
022550a0  00 00 18 00 00 00 03 00  4a 00 00 00 18 00 01 00  |........J.......|
022550b0  05 00 00 00 00 00 05 00  79 93 25 01 2c 87 d6 01  |........y.%.,...|
022550c0  79 93 25 01 2c 87 d6 01  79 93 25 01 2c 87 d6 01  |y.%.,...y.%.,...|
022550d0  79 93 25 01 2c 87 d6 01  00 40 00 00 00 00 00 00  |y.%.,....@......|
022550e0  00 40 00 00 00 00 00 00  06 00 00 00 00 00 00 00  |.@..............|
022550f0  04 03 24 00 4d 00 46 00  54 00 00 00 00 00 00 00  |..$.M.F.T.......|
02255100  80 00 00 00 48 00 00 00  01 00 40 00 00 00 06 00  |....H.....@.....|
02255110  00 00 00 00 00 00 00 00  3f 00 00 00 00 00 00 00  |........?.......|
02255120  40 00 00 00 00 00 00 00  00 00 04 00 00 00 00 00  |@...............|
02255130  00 00 04 00 00 00 00 00  00 00 04 00 00 00 00 00  |................|
02255140  21 40 55 21 00 00 00 00  b0 00 00 00 50 00 00 00  |!@U!........P...|
02255150  01 00 40 00 00 00 05 00  00 00 00 00 00 00 00 00  |..@.............|
02255160  01 00 00 00 00 00 00 00  40 00 00 00 00 00 00 00  |........@.......|
02255170  00 20 00 00 00 00 00 00  08 10 00 00 00 00 00 00  |. ..............|
02255180  08 10 00 00 00 00 00 00  21 01 54 21 21 01 d1 de  |........!.T!!...|
02255190  00 00 00 00 00 00 00 00  ff ff ff ff 00 00 00 00  |................|
022551a0  00 00 04 00 00 00 00 00  00 00 04 00 00 00 00 00  |................|
022551b0  00 00 04 00 00 00 00 00  21 40 55 21 00 00 00 00  |........!@U!....|
022551c0  b0 00 00 00 50 00 00 00  01 00 40 00 00 00 05 00  |....P.....@.....|
022551d0  00 00 00 00 00 00 00 00  01 00 00 00 00 00 00 00  |................|
022551e0  40 00 00 00 00 00 00 00  00 20 00 00 00 00 00 00  |@........ ......|
022551f0  08 10 00 00 00 00 00 00  08 10 00 00 00 00 02 00  |................|
```

# NTFS MTF Attributes

- The NTFS file system defines each object as a file along with a set of file attributes
- Elements such as the file's name, data, creation and modification time, and security information are file attributes
- Each attribute is denoted with a unique attribute type value and attribute name
- When a file's attributes and data can fit into the 1024 bytes of the MFT file record they are called resident

| MFT Attributes | |
|---|---|
| Name | Value (Hex) |
| $STANDARD_INFORMATION | 0x10 |
| $ATTRIBUTE_LIST | 0x20 |
| $FILE_NAME | 0x30 |
| $OBJECT_ID | 0x40 |
| $SECURITY_DESCRIPTOR | 0x50 |
| $VOLUME_NAME | 0x60 |
| $VOLUME_INFORMATION | 0x70 |
| $DATA | 0x80 |
| $INDEX_ROOT | 0x90 |
| $INDEX_ALLOCATION | 0xA0 |
| $BITMAP | 0xB0 |
| $REPARSE_POINT | 0xC0 |
| $EA_INFORMATION | 0xD0 |
| $EA | 0xE0 |
| $LOGGED_UTILITY_STREAM | 0x100 |

# NTFS Attributes

- If a file's attributes and data can not fit into the MFT file record they are called non-resident
- For non-resident attributes, it is necessary to allocated one or more clusters of disk space
- If a MFT record grows beyond the allocated 1024 bytes additional MFT records will be added and will generate an attribute list, which is also an attribute itself

# NTFS File Attribute Descriptions

- **$STANDARD_INFORMATION**
  - ✓ **Timestamp and link information**

- **$FILE_NAME**
  - ✓ **Information relative to short (8.3) or long (Unicode) filenames**
  - ✓ **Hard link information**

- **$DATA**
  - ✓ **Raw file data**

# NTFS File Attribute Descriptions

- **$ATTRIBUTE_LIST**
  - ✓ **Lists the location of all attribute records that do not fit in the MFT record.**
- **$SECURITY_DESCRIPTOR**
  - ✓ **File ownership and access information**
- **$OBJECT_ID**
  - ✓ **Provides a global tracking identifier for files**

# NTFS File Attribute Descriptions

- **$REPARSE_POINT**
  - ✓ **Used for volume mount points**

- **$INDEX_ROOT, $INDEX_ALLOCATION, $BITMAP**
  - ✓ **Used to implement folders and other indexes**

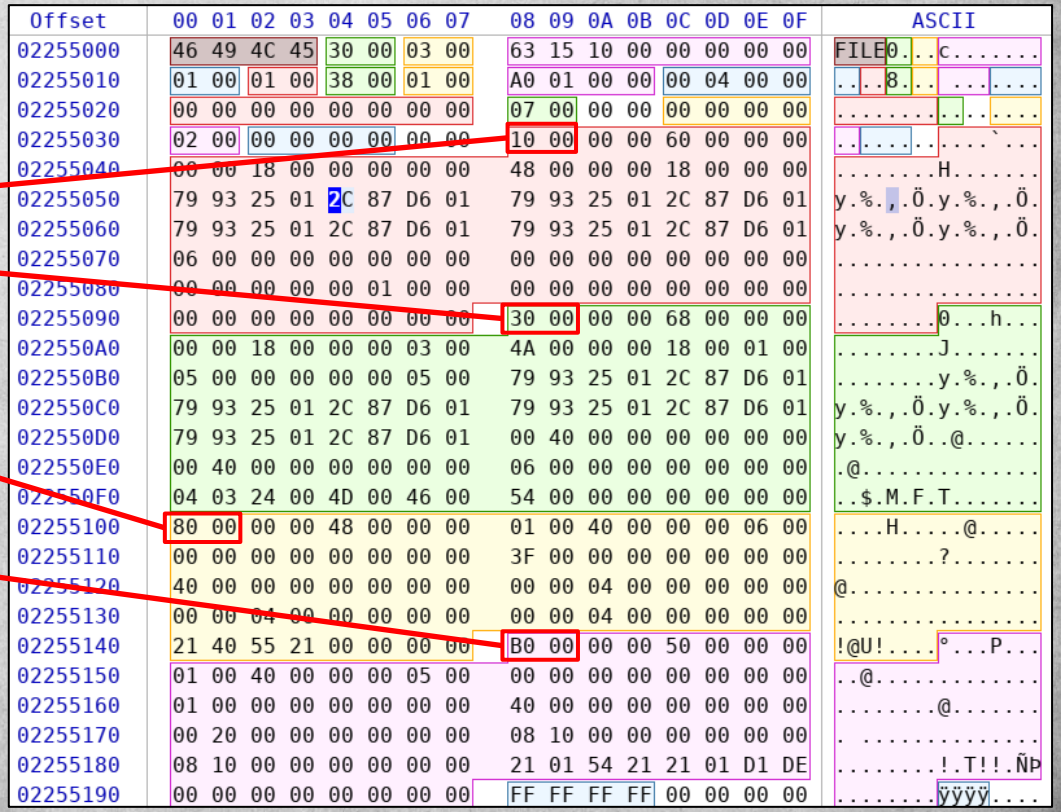
- **$VOLUME_INFORMATION**
  - ✓ **Volume version information**

- **$VOLUME_NAME**
  - ✓ **Volume label information**

# MFT Attribute Type Identifiers

- **What file attributes does the $MFT entry have?**

| MFT Attributes | |
|---|---|
| Name | Value (Hex) |
| $STANDARD_INFORMATION | 0x10 |
| $ATTRIBUTE_LIST | 0x20 |
| $FILE_NAME | 0x30 |
| $OBJECT_ID | 0x40 |
| $SECURITY_DESCRIPTOR | 0x50 |
| $VOLUME_NAME | 0x60 |
| $VOLUME_INFORMATION | 0x70 |
| $DATA | 0x80 |
| $INDEX_ROOT | 0x90 |
| $INDEX_ALLOCATION | 0xA0 |
| $BITMAP | 0xB0 |
| $REPARSE_POINT | 0xC0 |
| $EA_INFORMATION | 0xD0 |
| $EA | 0xE0 |
| $LOGGED_UTILITY_STREAM | 0x100 |



# $STANDARD_INFORMATION

# $STANDARD_INFORMATION Attribute

- **The $SI attribute exists for every file and directory and is always resident**
- **$SI attribute identifier: 0x10**
- **The elements of the $SI attribute are non-essential and include:**

- ✓ **Creation Time**
- ✓ **File Altered Time**
- ✓ **MFT Altered Time**
- ✓ **File Accessed Time**
- ✓ **Flags**
- ✓ **Maximum Versions**
- ✓ **Version Number**
- ✓ **Class ID**
- ✓ **Owner ID**
- ✓ **Security ID**
- ✓ **Quota**
- ✓ **Update Sequence Number**

# $STANDARD_INFORMATION Flags

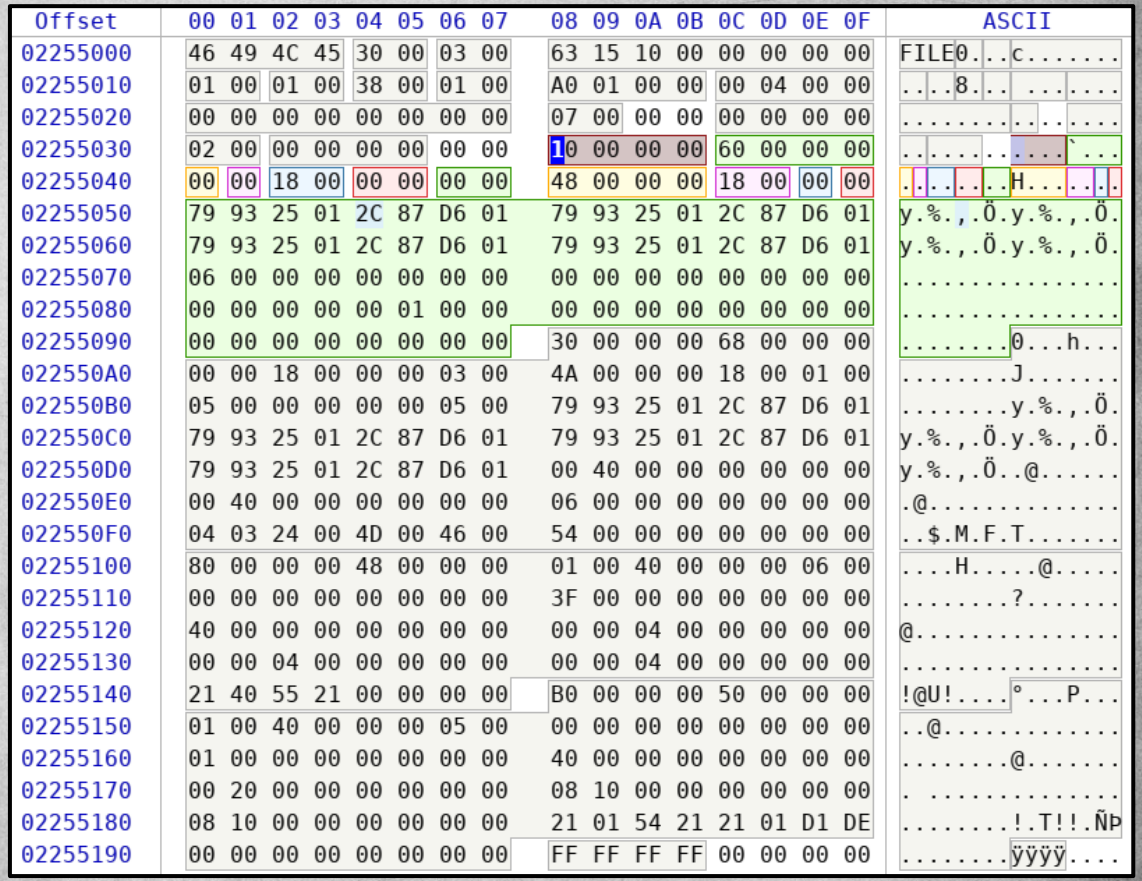
- **$SI flags are non-essential metadata but include the following values:**

- ✓ **0x0001 – Read Only**
- ✓ **0x0002 – Hidden**
- ✓ **0x0004 – System**
- ✓ **0x0020 – Archive**
- ✓ **0x0040 – Device**
- ✓ **0x0080 – Normal**
- ✓ **0x0100 – Temporary**
- ✓ **0x0200 – Sparse File**
- ✓ **0x0400 – Reparse Point**
- ✓ **0x0800 – Compressed**
- ✓ **0x1000 – Offline**
- ✓ **0x2000 – Non-Indexed**
- ✓ **0x4000 - Encrypted**

# $STANDARD_INFORMATION – 0x10

| $SI Header | | |
|---|---|---|
| Description | Offset | Bytes |
| Attribute | 0x38 | 4 |
| Total Length | 0x3C | 4 |
| Non-Resident Flag | 0x40 | 1 |
| Name Length | 0x41 | 1 |
| Name Offset | 0x42 | 2 |
| $SI Flags | 0x44 | 2 |
| Attribute ID | 0x46 | 2 |
| Attribute Length | 0x48 | 4 |
| Attribute Data Offset | 0x4C | 2 |
| Index Flag | 0x4E | 1 |
| Padding | 0x4F | 1 |

| $SI | | |
|---|---|---|
| Description | Offset | Bytes |
| File Creation Time | 0x50 | 8 |
| File Altered Time | 0x58 | 8 |
| Record Changed Time | 0x60 | 8 |
| Last Access Time | 0x68 | 8 |
| File Permissions | 0x70 | 4 |
| Maximum Versions | 0x74 | 4 |
| Version Number | 0x78 | 4 |
| Class ID | 0x7C | 4 |
| Owner ID | 0x80 | 4 |
| Security ID | 0x84 | 4 |
| Quota Charged | 0x88 | 8 |
| Update Sequence Number (USN) | 0x90 | 8 |



# $FILE_NAME

# $FILE_NAME Attribute

- **The $FN attribute is used to:**
  - ✓ **Stores the file name and parent directory of each file in the MFT entry**
  - ✓ **Store data in a directory index**

- **$FN attribute identifier: 0x30**

- **The elements of $FN attribute include:**

  - ✓ **File Reference or Parent Directory - N**
  - ✓ **File Creation Time - N**
  - ✓ **File Modification Time - N**
  - ✓ **MFT Modification Time - N**
  - ✓ **File Access Time - N**
  - ✓ **Allocates Size of File - N**
  - ✓ **Real Size of File - N**
  - ✓ **Flags - N**
  - ✓ **Reparse Value - N**
  - ✓ **Length of Name – Y for Directory Index**
  - ✓ **Namespace - Y for Directory Index**
  - ✓ **Name - Y for Directory Index**

# $FILE_NAME Flags

- **$FN flags:**

| $FN Flag Value | $FN Flag |
|---|---|
| 0x0001 | Read Only |
| 0x0002 | Hidden |
| 0x0004 | System |
| 0x0020 | Archive |
| 0x0040 | Device |
| 0x0080 | Normal |
| 0x0100 | Temporary |
| 0x0200 | Sparse File |
| 0x0400 | Reparse Point |
| 0x0800 | Compressed |
| 0x1000 | Offline |
| 0x2000 | Non-Indexed |
| 0x4000 | Encrypted |

# $FILE_NAME – 0x30

| $FN Header | | |
|---|---|---|
| Description | Offset | Bytes |
| Attribute | 0x98 | 4 |
| Total Length | 0x9C | 4 |
| Non-Resident Flag | 0xA0 | 1 |
| Name Length | 0xA1 | 1 |
| Name Offset | 0xA2 | 2 |
| $FN Flags | 0xA4 | 2 |
| Attribute ID | 0xA6 | 2 |
| Attribute Length | 0xA8 | 4 |
| Attribute Data Offset | 0xAC | 2 |
| Index Flag | 0xAE | 1 |
| Padding | 0xAF | 1 |

| $FN | | |
|---|---|---|
| Description | Offset | Bytes |
| Parent Directory Record # | 0xB0 | 6 |
| Parent Directory Sequence # | 0xB6 | 2 |
| File Creation Time | 0xB8 | 8 |
| File Modification Time | 0xC0 | 8 |
| Record Modification Time | 0xC8 | 8 |
| Last Accessed Time | 0xD0 | 8 |
| Attribute Allocated Size | 0xD8 | 8 |
| Attribute Actual Size | 0xE0 | 8 |
| $FN Flags | 0xE8 | 4 |
| EA and Reparse | 0xEC | 4 |
| Filename Length | 0xF0 | 1 |
| Filename Namespace | 0xF1 | 1 |
| File Name | 0xF2 | 8 |

| Offset | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | ASCII |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 02255000 | 46 | 49 | 4C | 45 | 30 | 00 | 03 | 00 | 63 | 15 | 10 | 00 | 00 | 00 | 00 | 00 | FILE0..c..... |
| 02255010 | 01 | 00 | 01 | 00 | 38 | 00 | 01 | 00 | A0 | 01 | 00 | 00 | 00 | 04 | 00 | 00 | ..8.. ....... |
| 02255020 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 07 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ..... ..... |
| 02255030 | 02 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 10 | 00 | 00 | 00 | 60 | 00 | 00 | 00 | ..... ....`... |
| 02255040 | 00 | 00 | 18 | 00 | 00 | 00 | 00 | 00 | 48 | 00 | 00 | 00 | 18 | 00 | 00 | 00 | ..... H..... |
| 02255050 | 79 | 93 | 25 | 01 | 2C | 87 | D6 | 01 | 79 | 93 | 25 | 01 | 2C | 87 | D6 | 01 | y.%.,Ö.y.%.,Ö. |
| 02255060 | 79 | 93 | 25 | 01 | 2C | 87 | D6 | 01 | 79 | 93 | 25 | 01 | 2C | 87 | D6 | 01 | y.%.,Ö.y.%.,Ö. |
| 02255070 | 06 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ..... |
| 02255080 | 00 | 00 | 00 | 00 | 00 | 01 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ..... |
| 02255090 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 30 | 00 | 00 | 00 | 68 | 00 | 00 | 00 | ..... 0...h... |
| 022550A0 | 00 | 00 | 18 | 00 | 00 | 00 | 03 | 00 | 4A | 00 | 00 | 00 | 18 | 00 | 01 | 00 | ..... J..... |
| 022550B0 | 05 | 00 | 00 | 00 | 00 | 00 | 05 | 00 | 79 | 93 | 25 | 01 | 2C | 87 | D6 | 01 | ..... y.%.,Ö. |
| 022550C0 | 79 | 93 | 25 | 01 | 2C | 87 | D6 | 01 | 79 | 93 | 25 | 01 | 2C | 87 | D6 | 01 | y.%.,Ö.y.%.,Ö. |
| 022550D0 | 79 | 93 | 25 | 01 | 2C | 87 | D6 | 01 | 00 | 40 | 00 | 00 | 00 | 00 | 00 | 00 | y.%.,Ö..@..... |
| 022550E0 | 00 | 40 | 00 | 00 | 00 | 00 | 00 | 00 | 06 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | .@..... |
| 022550F0 | 04 | 03 | 24 | 00 | 4D | 00 | 46 | 00 | 54 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ..$.M.F.T..... |
| 02255100 | 80 | 00 | 00 | 00 | 48 | 00 | 00 | 00 | 01 | 00 | 40 | 00 | 00 | 00 | 06 | 00 | ....H.....@..... |
| 02255110 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 3F | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ..... ?..... |
| 02255120 | 40 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 04 | 00 | 00 | 00 | 00 | 00 | @..... |
| 02255130 | 00 | 00 | 04 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 04 | 00 | 00 | 00 | 00 | 00 | ..... |
| 02255140 | 21 | 40 | 55 | 21 | 00 | 00 | 00 | 00 | B0 | 00 | 00 | 00 | 50 | 00 | 00 | 00 | !@U!....°...P... |
| 02255150 | 01 | 00 | 40 | 00 | 00 | 00 | 05 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ..@..... |
| 02255160 | 01 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 40 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ..... @..... |
| 02255170 | 00 | 20 | 00 | 00 | 00 | 00 | 00 | 00 | 08 | 10 | 00 | 00 | 00 | 00 | 00 | 00 | . ..... |
| 02255180 | 08 | 10 | 00 | 00 | 00 | 00 | 00 | 00 | 21 | 01 | 54 | 21 | 21 | 01 | D1 | DE | ..... !.T!!.Ñþ |
| 02255190 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | FF | FF | FF | FF | 00 | 00 | 00 | 00 | ..... ÿÿÿÿ.... |

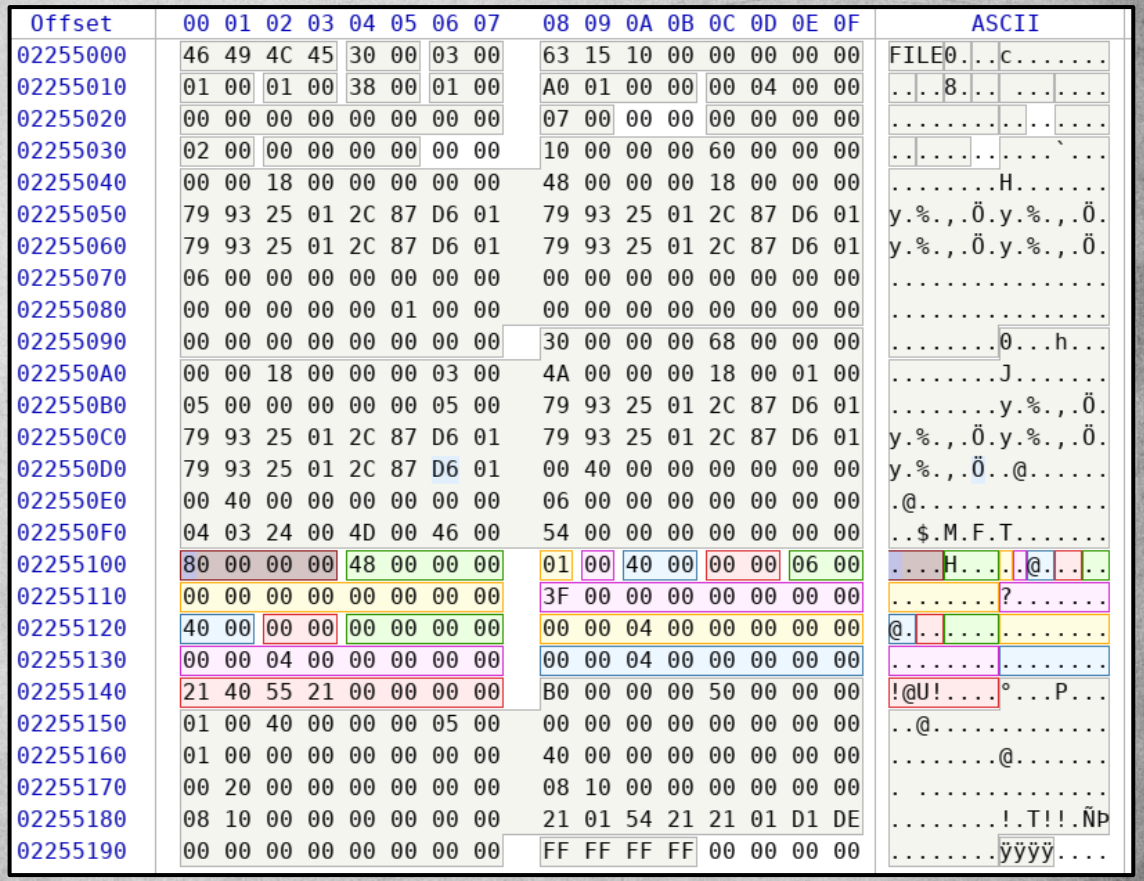
# $DATA

# $DATA Attribute

- The $DATA attribute does not have a structure
- After the header, the remaining data corresponds to raw file data
- $DATA attribute identifier: 0x80
- $DATA has no minimum or maximum size
- Raw data over 700 bytes in size is non-resident
- For most files, the $DATA attribute is the last attribute in the MFT entry

# $DATA – 0x80

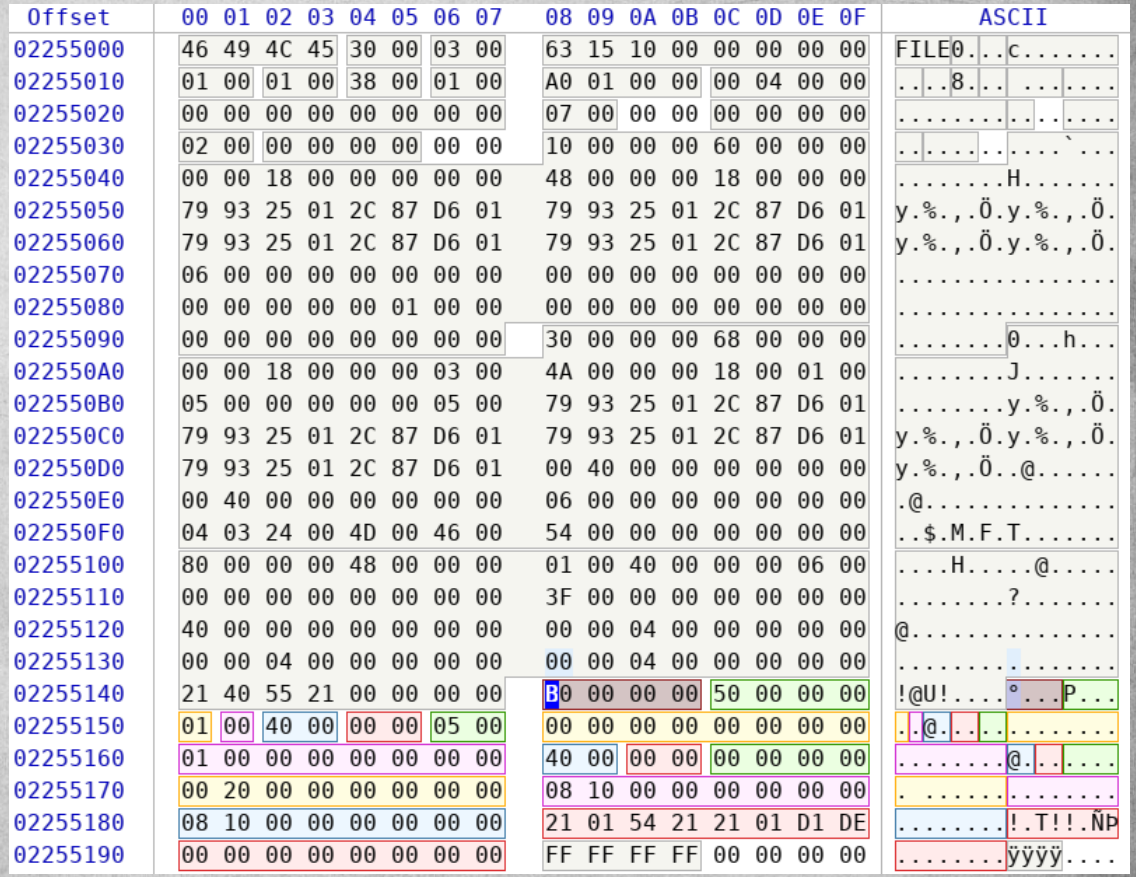
| $DATA Header | | |
|---|---|---|
| **Description** | **Offset** | **Bytes** |
| Attribute | 0x100 | 4 |
| Total Length | 0x104 | 4 |
| Non-Resident Flag | 0x108 | 1 |
| Name Length | 0x109 | 1 |
| Name Offset | 0x10A | 2 |
| $DATA Flags | 0x10C | 2 |
| Attribute ID | 0x10E | 2 |
| 1st Virtual Cluster # (VCN) | 0x110 | 8 |
| Last Virtual Cluster # (VCN) | 0x118 | 8 |
| Data Runs Offset | 0x120 | 2 |
| Compression Unit Size | 0x122 | 2 |
| Padding | 0x124 | 4 |
| Attribute Allocated Size | 0x128 | 8 |
| Attribute Actual Size | 0x130 | 8 |
| Attribute Initialized Size | 0x138 | 8 |
| $DATA | 0x140 | 8 |



# $BITMAP

# $DATA – 0xB0

| $BITMAP Header | | |
|---|---|---|
| Description | Offset | Bytes |
| Attribute | 0x148 | 4 |
| Total Length | 0x14C | 4 |
| Non-Resident Flag | 0x150 | 1 |
| Name Length | 0x151 | 1 |
| Name Offset | 0x152 | 2 |
| $BTIMAP Flags | 0x154 | 2 |
| Attribute ID | 0x156 | 2 |
| 1st Virtual Cluster # (VCN) | 0x158 | 8 |
| Last Virtual Cluster # (VCN) | 0x160 | 8 |
| Data Runs Offset | 0x168 | 2 |
| Compression Unit Size | 0x16A | 2 |
| Padding | 0x16C | 4 |
| Attribute Allocated Size | 0x170 | 8 |
| Attribute Actual Size | 0x178 | 8 |
| Attribute Initialized Size | 0x180 | 8 |
| $BITMAP | 0x188 | 8 |



# NTFS Data Recovery Process

# NTFS Data Recovery Process

**When attempting to recover data on an NTFS partition the following steps will help organize the recovery process:**

- ✓ **Determine NTFS Partition Offset**
- ✓ **Analyze NTFS Master Boot Record**
- ✓ **Identify System Generated Master File Table Entries**
- ✓ **Identify User Generated Master File Table Entries**
- ✓ **Classify Files as Existing or Deleted**
- ✓ **Classify Files as Resident or Non-Resident**
- ✓ **Locate File Contents**
- ✓ **Recover File Contents**

# Determine NTFS Partition Offset

```
Forensics $ sudo fdisk -l disk2.dd
Disk disk2.dd: 240.4 MiB, 252051456 bytes, 492288 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disklabel type: dos
Disk identifier: 0x65e468c4

Device     Boot Start    End Sectors  Size Id Type
disk2.dd1        2048 206847  204800  100M  7 HPFS/NTFS/exFAT
```

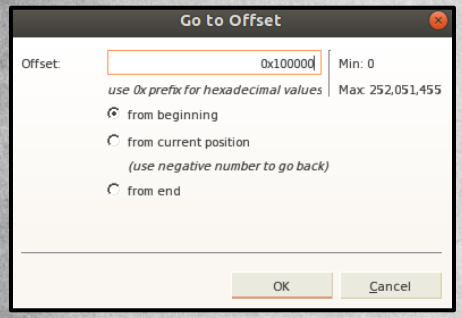
**First NTFS Sector: 0x0800**

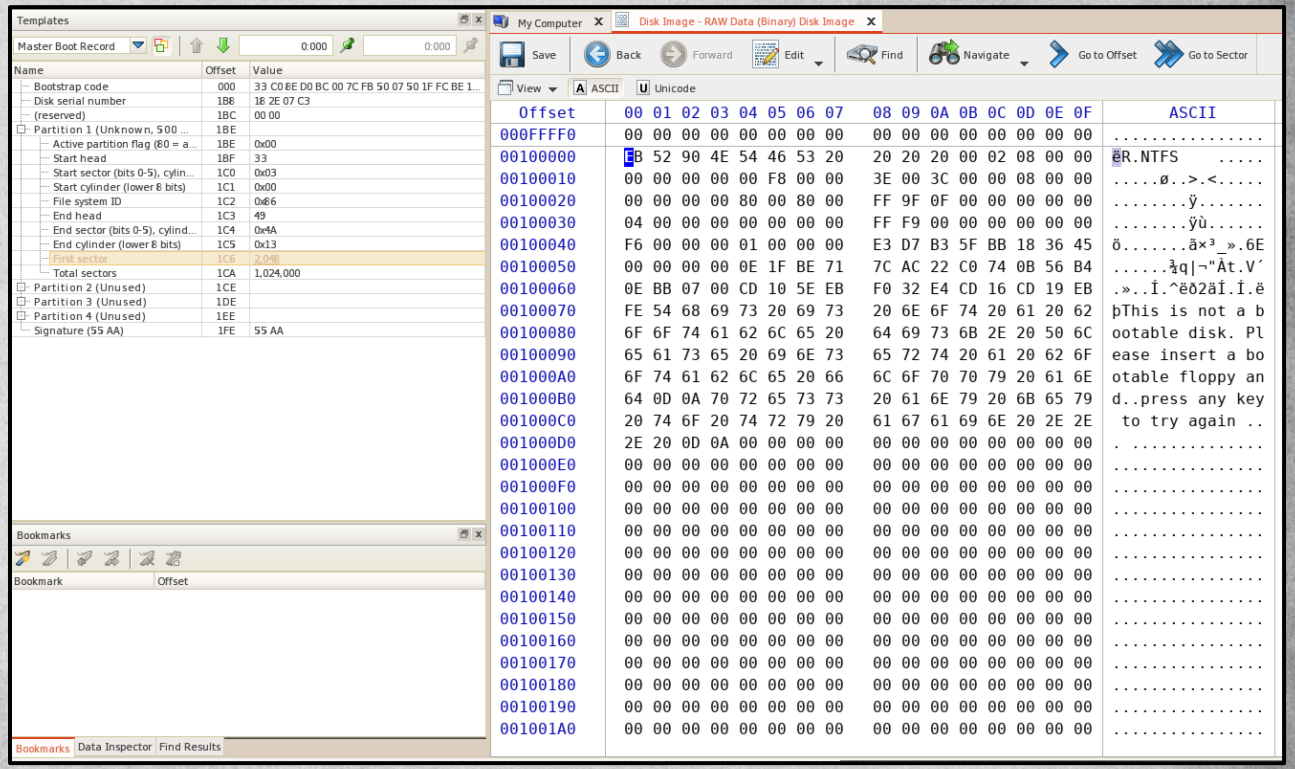
**Sector Size: 0x0200**

$$0x0800 \times 0x0200 = 0x100000$$

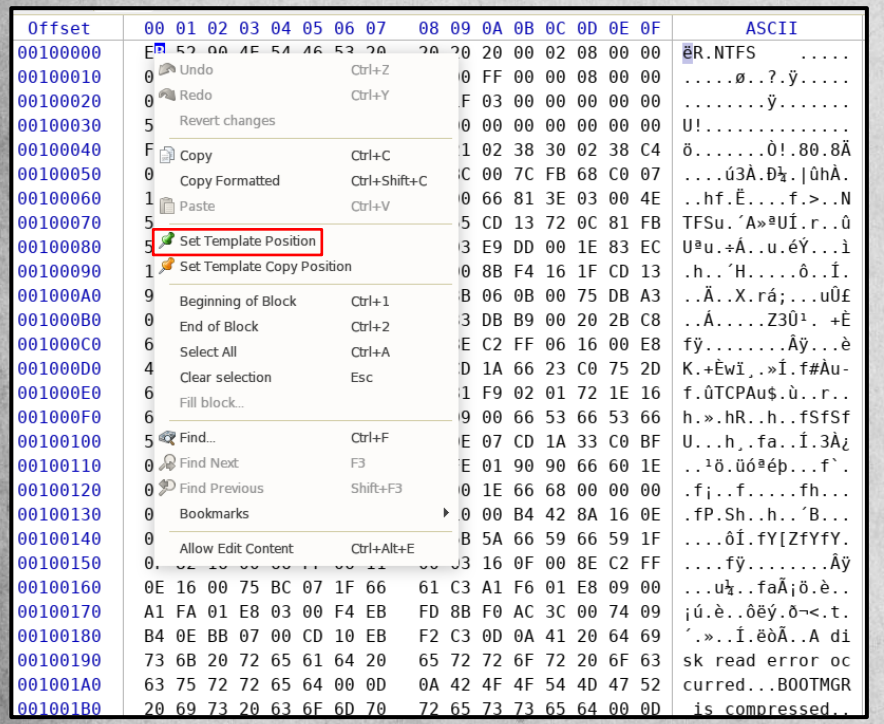
**The NTFS Partition starts at address 0x100000**

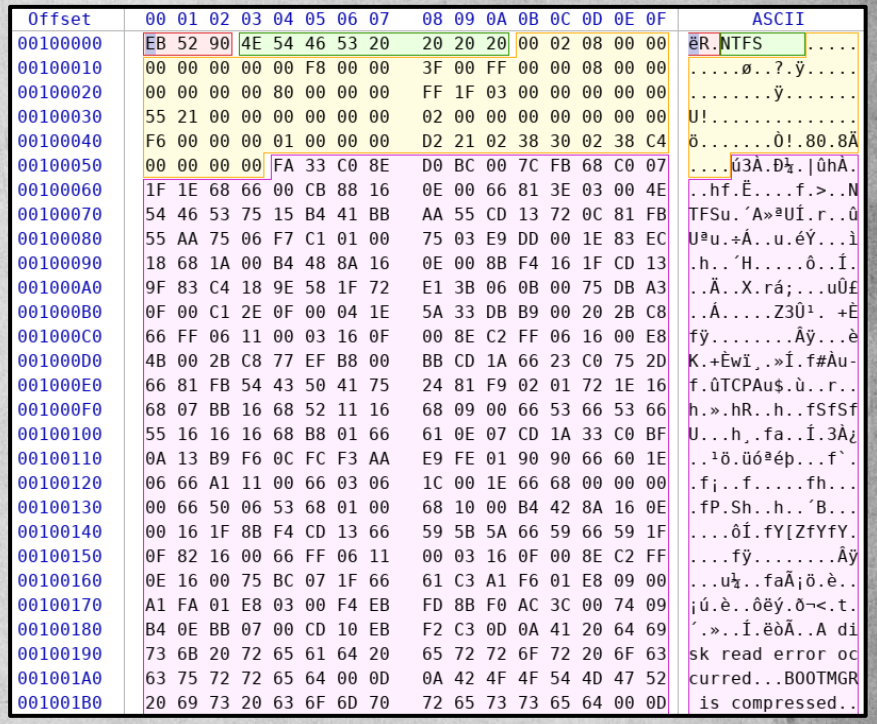
# Move to Start Address of NTFS Partition



# Setting Templates

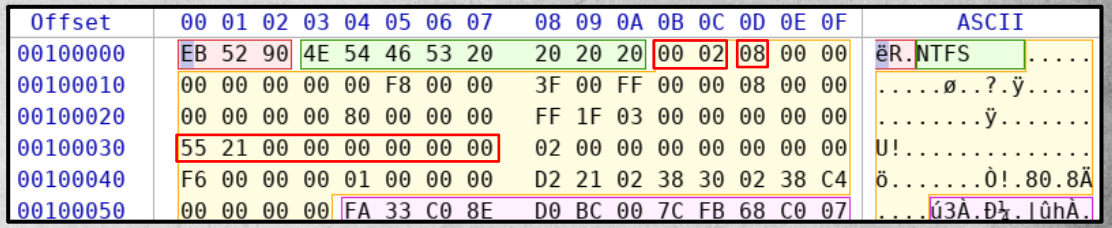


# Analyze NTFS Master Boot Record

**The $MFT start address is from start of the disk partition!**

**Bytes / Sector => 0x200**
**Sectors / Cluster => 0x8**
**Bytes / Cluster => 0x1000**
**Disk Offset => 0x100000**
**$MFT Cluster Number => 0x2155**



| Offset | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | ASCII |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 00100000 | EB | 52 | 90 | 4E | 54 | 46 | 53 | 20 | 20 | 20 | 20 | 00 | 02 | 08 | 00 | 00 | ëR.NTFS .... |
| 00100010 | 00 | 00 | 00 | 00 | 00 | F8 | 00 | 00 | 3F | 00 | FF | 00 | 00 | 08 | 00 | 00 | .....ø..?.ÿ..... |
| 00100020 | 00 | 00 | 00 | 00 | 80 | 00 | 00 | 00 | FF | 1F | 03 | 00 | 00 | 00 | 00 | 00 | ........ÿ....... |
| 00100030 | 55 | 21 | 00 | 00 | 00 | 00 | 00 | 00 | 02 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | U!.............. |
| 00100040 | F6 | 00 | 00 | 00 | 01 | 00 | 00 | 00 | D2 | 21 | 02 | 38 | 30 | 02 | 38 | C4 | ö.......Ò!.80.8Ä |
| 00100050 | 00 | 00 | 00 | 00 | FA | 33 | C0 | 8E | D0 | BC | 00 | 7C | FB | 68 | C0 | 07 | ....ú3À.Ð¼.\|ûhÀ. |

**$MFT Location (Bytes)**

**0x2155**
**✖ 0x1000**

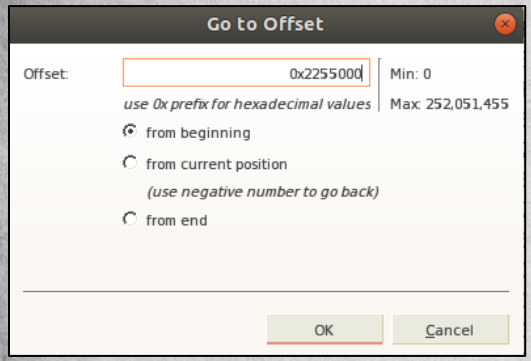
**0x2155000**

**$MFT Location + Disk Offset (Bytes)**

**0x2155000**
**✚ 0x100000**

**0x2255000**

# System Generated MFT Entries




| Offset | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | ASCII |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 02255000 | 46 | 49 | 4C | 45 | 30 | 00 | 03 | 00 | 63 | 15 | 10 | 00 | 00 | 00 | 00 | 00 | FILE0..c..... |
| 02255010 | 01 | 00 | 01 | 00 | 38 | 00 | 01 | 00 | A0 | 01 | 00 | 00 | 00 | 04 | 00 | 00 | ..8. ....  |
| 02255020 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 07 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ..... |
| 02255030 | 02 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 10 | 00 | 00 | 00 | 60 | 00 | 00 | 00 | ....`... |
| 02255040 | 00 | 00 | 18 | 00 | 00 | 00 | 00 | 00 | 48 | 00 | 00 | 00 | 18 | 00 | 00 | 00 | .....H..... |
| 02255050 | 79 | 93 | 25 | 01 | 2C | 87 | D6 | 01 | 79 | 93 | 25 | 01 | 2C | 87 | D6 | 01 | y.%.,.Ö.y.%.,.Ö. |
| 02255060 | 79 | 93 | 25 | 01 | 2C | 87 | D6 | 01 | 79 | 93 | 25 | 01 | 2C | 87 | D6 | 01 | y.%.,.Ö.y.%.,.Ö. |
| 02255070 | 06 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ..... |
| 02255080 | 00 | 00 | 00 | 00 | 00 | 01 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ..... |
| 02255090 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 30 | 00 | 00 | 00 | 68 | 00 | 00 | 00 | .....0...h... |
| 022550A0 | 00 | 00 | 18 | 00 | 00 | 00 | 03 | 00 | 4A | 00 | 00 | 00 | 18 | 00 | 01 | 00 | .....J..... |
| 022550B0 | 05 | 00 | 00 | 00 | 00 | 00 | 05 | 00 | 79 | 93 | 25 | 01 | 2C | 87 | D6 | 01 | .....y.%.,.Ö. |
| 022550C0 | 79 | 93 | 25 | 01 | 2C | 87 | D6 | 01 | 79 | 93 | 25 | 01 | 2C | 87 | D6 | 01 | y.%.,.Ö.y.%.,.Ö. |
| 022550D0 | 79 | 93 | 25 | 01 | 2C | 87 | D6 | 01 | 00 | 40 | 00 | 00 | 00 | 00 | 00 | 00 | y.%.,.Ö..@..... |
| 022550E0 | 00 | 40 | 00 | 00 | 00 | 00 | 00 | 00 | 06 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | .@..... |
| 022550F0 | 04 | 03 | 24 | 00 | 4D | 00 | 46 | 00 | 54 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ..$.M.F.T..... |
| 02255100 | 80 | 00 | 00 | 00 | 48 | 00 | 00 | 00 | 01 | 00 | 40 | 00 | 00 | 00 | 06 | 00 | ....H.....@..... |
| 02255110 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 3F | 00 | 00 | 00 | 00 | 00 | 00 | 00 | .....?..... |
| 02255120 | 40 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 04 | 00 | 00 | 00 | 00 | 00 | @..... |
| 02255130 | 00 | 00 | 04 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 04 | 00 | 00 | 00 | 00 | 00 | ..... |
| 02255140 | 21 | 40 | 55 | 21 | 00 | 00 | 00 | 00 | B0 | 00 | 00 | 00 | 50 | 00 | 00 | 00 | !@U!....°...P... |
| 02255150 | 01 | 00 | 40 | 00 | 00 | 00 | 05 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ..@..... |
| 02255160 | 01 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 40 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | .....@..... |
| 02255170 | 00 | 20 | 00 | 00 | 00 | 00 | 00 | 00 | 08 | 10 | 00 | 00 | 00 | 00 | 00 | 00 | . ..... |
| 02255180 | 08 | 10 | 00 | 00 | 00 | 00 | 00 | 00 | 21 | 01 | 54 | 21 | 21 | 01 | D1 | DE | .....!.T!!.ÑÞ |
| 02255190 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | FF | FF | FF | FF | 00 | 00 | 00 | 00 | .....ÿÿÿÿ.... |

# System Generated MFT Entries

- **The information provided in the MFT provides everything necessary to rebuild digital artifacts in an NFTS file system**
- **MFT Facts**
  - ✓ **Each MFT entry is 1024 bytes**
  - ✓ **The first 39 MFT file entries are system generated***
    - **$MFT starts at entry 0**
    - **WPSettings.dat ends at entry 38**
  - ✓ **The 39th entry of the MFT table is the start of user generated data**

# MFT System Files

- **MFT System Files:**

- ✓ $MFT – 0
- ✓ $MFTMirr – 1
- ✓ $LogFile – 2
- ✓ $Volume – 3
- ✓ $AttrDef – 4
- ✓ $I30 – 5
- ✓ $Bitmap – 6
- ✓ $Boot – 7
- ✓ $BadClus – 8
- ✓ $Secure – 9
- ✓ $UpCase – 10
- ✓ $Extend – 11
- ✓ $MFT Extensions – 12 – 23
- ✓ $Extend\$Quota – 24
- ✓ $Extend\$ObjId – 25
- ✓ $Extend\$Reparse – 26
- ✓ $RMMetadata – 27
- ✓ $Repair – 28
- ✓ $Deleted – 29
- ✓ $TxtLog – 30
- ✓ $Txf – 31
- ✓ $Tops – 32
- ✓ $TxfLog – 33
- ✓ $TxfLogContainer1 – 34
- ✓ $TxfLogContainer2 – 35
- ✓ System Volume Information – 36
- ✓ Indexer Volume GUID – 37
- ✓ WPSettings.dat – 38

# MFT System File Definitions

- $MFT
  - ✓ MFT Entry 0
  - ✓ Provides definitions for all files on the partition
  - ✓ $MFT has a unique attribute called $BITMAP which is used to manage MFT entries
- $MFTMirr
  - ✓ MFT Entry 1
  - ✓ Backup of the 1st 4 $MFT entries
- $LogFile
  - ✓ MFT Entry 2
  - ✓ Used as the NTFS journal which tracks changes to system metadata
- $Volume
  - ✓ MFT Entry 3
  - ✓ Contains information specific to the partition including volume label, volume identifier, file system version, and volume flags

# MFT System File Definitions

- $AttrDef
  - ✓ MFT Entry 4
  - ✓ Tracks all file system attribute names and identifiers
- $I30
  - ✓ MFT Entry 5
  - ✓ Also known as the root directory
  - ✓ A key file for recovering deleted and overwritten data in NTFS
- $Bitmap
  - ✓ MFT Entry 6
  - ✓ Tracks cluster utilization
- $Boot
  - ✓ MFT Entry 7
  - ✓ Contains the boot sector and boot code in its $DATA attribute
  - ✓ $Boot always starts in sector 0
  - ✓ $Boot is the only file that cannot be relocated

# MFT System File Definitions

- $BadClus
  - ✓ MFT Entry 8
  - ✓ Lists all bad clusters in the partition
- $Secure
  - ✓ MFT Entry 9
  - ✓ Contains all security descriptors for all files on the partition
- $UpCase
  - ✓ MFT Entry 10
  - ✓ Converts lowercase characters to matching Unicode uppercase characters
  - ✓ The purpose is to provide proper formatting for optional extensions including quotas, reparse point data, and identifiers
- $Extend
  - ✓ MFT Entry 11
  - ✓ Extended data to aid with disk quotas, reparse point data, and identifiers

# MFT System File Definitions

- **$Extend\$Quota**
  - ✓ **MFT Entry 24**
  - ✓ **Contains user assigned quota limits on the volume space**
- **$Extend\$ObjId**
  - ✓ **MFT Entry 25**
  - ✓ **Contains all file object identification numbers**
- **$Extend\$Reparse**
  - ✓ **MFT Entry 26**
  - ✓ **Contains information about files and folders on the volume include reparse point data**

# NTFS User Generated Data

# Finding User Generated Data

- Let’s apply what we know so far "disk2.dd":
  - ✓ Location of the disk boot sector
    - 0x0
  - ✓ Location of NTFS boot sector
    - 0x100000
  - ✓ Location of the MFT
    - 0x2255000
  - ✓ The number of system level MFT entries
    - 39
    - We know that each MFT entry is 1024 bytes
    - So from the start of MFT, we must add 9C00

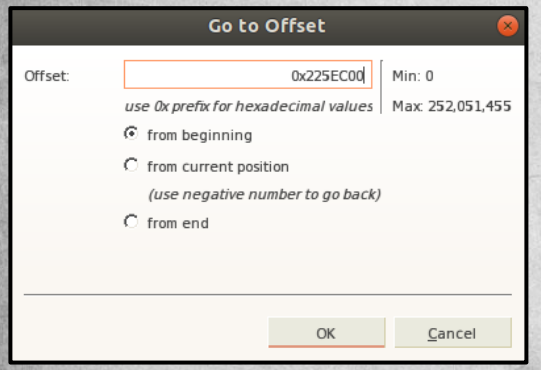
39 * 1024 = 39,936

39,936 => 9C00

0x2255000 + 0x9C00 = **0x225EC00**

User generated data offset

# User Generated Data




| Offset | 00 | 01 | 02 | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A | 0B | 0C | 0D | 0E | 0F | ASCII |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0225EC00 | 46 | 49 | 4C | 45 | 30 | 00 | 03 | 00 | C2 | 2E | 10 | 00 | 00 | 00 | 00 | 00 | FILE0...Â..... |
| 0225EC10 | 01 | 00 | 01 | 00 | 38 | 00 | 01 | 00 | 58 | 01 | 00 | 00 | 00 | 04 | 00 | 00 | ....8...X..... |
| 0225EC20 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 03 | 00 | 00 | 00 | 27 | 00 | 00 | 00 | ............'... |
| 0225EC30 | 02 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 10 | 00 | 00 | 00 | 60 | 00 | 00 | 00 | ..........`... |
| 0225EC40 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 48 | 00 | 00 | 00 | 18 | 00 | 00 | 00 | ........H..... |
| 0225EC50 | 3C | 94 | 23 | 21 | 2C | 87 | D6 | 01 | DE | FF | 77 | C3 | 28 | 87 | D6 | 01 | <.#!,.Ö.ÞÿwÃ(.Ö. |
| 0225EC60 | CC | 83 | A1 | C3 | 28 | 87 | D6 | 01 | 3C | 94 | 23 | 21 | 2C | 87 | D6 | 01 | Ì.¡Ã(.Ö.<.#!,.Ö. |
| 0225EC70 | 20 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ................ |
| 0225EC80 | 00 | 00 | 00 | 00 | 08 | 01 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ................ |
| 0225EC90 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 30 | 00 | 00 | 00 | 70 | 00 | 00 | 00 | ........0...p... |
| 0225ECA0 | 00 | 00 | 00 | 00 | 00 | 00 | 02 | 00 | 58 | 00 | 00 | 00 | 18 | 00 | 01 | 00 | ........X....... |
| 0225ECB0 | 05 | 00 | 00 | 00 | 00 | 00 | 05 | 00 | 3C | 94 | 23 | 21 | 2C | 87 | D6 | 01 | ........<.#!,.Ö. |
| 0225ECC0 | 3C | 94 | 23 | 21 | 2C | 87 | D6 | 01 | 3C | 94 | 23 | 21 | 2C | 87 | D6 | 01 | <.#!,.Ö.<.#!,.Ö. |
| 0225ECD0 | 3C | 94 | 23 | 21 | 2C | 87 | D6 | 01 | 00 | A0 | 00 | 00 | 00 | 00 | 00 | 00 | <.#!,.Ö.. ...... |
| 0225ECE0 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 20 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ........ ....... |
| 0225ECF0 | 0B | 00 | 4D | 00 | 69 | 00 | 6E | 00 | 69 | 00 | 6F | 00 | 6E | 00 | 73 | 00 | ..M.i.n.i.o.n.s. |
| 0225ED00 | 2E | 00 | 4A | 00 | 50 | 00 | 47 | 00 | 80 | 00 | 00 | 00 | 48 | 00 | 00 | 00 | ..J.P.G.....H... |
| 0225ED10 | 01 | 00 | 00 | 00 | 00 | 00 | 01 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ................ |
| 0225ED20 | 09 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 40 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ........@....... |
| 0225ED30 | 00 | A0 | 00 | 00 | 00 | 00 | 00 | 00 | 21 | 91 | 00 | 00 | 00 | 00 | 00 | 00 | . ......!....... |
| 0225ED40 | 21 | 91 | 00 | 00 | 00 | 00 | 00 | 00 | 21 | 0A | 88 | 05 | 00 | 00 | 00 | 00 | !.......!....... |
| 0225ED50 | FF | FF | FF | FF | 82 | 79 | 47 | 11 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | 00 | ÿÿÿÿ.yG......... |

# Additional User Generated Data

- **Since $MFT records are generated for every file within NTFS and are 1024 bytes per record, you can use the same process by adding this offset through the rest of the disk**
- **This results in 4 user generated files**
  - ✓ **Minions.jpg**
  - ✓ **Simple.txt**
  - ✓ **Meow.jpg**
  - ✓ **Rabbidz.jpg**
- **Now that MFT records have been found for each user generated file, the next step is to use the contents in the MFT to find file contents and recover the data**

# References


- File System Forensic Analysis, Carrier, 2005
- NTFS
  - http://ntfs.com/ntfs_basics.htm
- Unicode
  - http://www.unicode.org/standard/WhatIsUnicode.html