

# Forensic Analysis of NTFS Metadata Artifacts for Deleted File Reconstruction

Suparshva Jain

School Of Technology And Engineering  
Sushant University  
Gurugram, Haryana, India

Ayush Pandey

School Of Technology And Engineering  
Sushant University  
Gurugram, Haryana, India

Ms. Shiksha Kumari

School Of Technology And Engineering  
Sushant University  
Gurugram, Haryana, India

**Abstract**—Digital forensic investigators face challenges in gathering evidence due to the increased use of anti-forensic techniques, particularly with regard to intentional data deletion in the New Technology File System (NTFS). This research will provide a detailed forensic analysis of the NTFS metadata artifacts and will examine the recovery of files that have been permanently erased from an NTFS filesystem. During the 12-week experimental period, an empirically-controlled virtualized laboratory was created using VMware and VirtualBox to isolate the forensic residuals left on a Windows 10 guest OS, and to conduct an analysis of these residuals. The experimental study will look primarily at the persistence of the metadata stored within the Master File Table (\$MFT) and the underlying cluster allocation logic of the \$Bitmap file, as well as at the transactional journaling contained in the \$LogFile. During the course of the experiment, all files on a test partition (10 GB) were permanently deleted using “Shift+Delete” key combinations and the Recycle Bin was emptied. Bitstream forensic images in .E01 format were captured during the deletion process for further analysis at a low-level. The methodology included automated parsing of the files using Autopsy and manual hex-level inspection using The Sleuth Kit (TSK) and HxD.

**Index Terms**—Digital forensics, file reconstruction, Master File Table, metadata artifacts, NTFS architecture.

## I. INTRODUCTION

Reconstructing the sequence of events from non-volatile media is one of the primary tenets of digital investigation in the present day. With storage capacities increasing and file systems becoming markedly more intricate, the sheer number of data items generated by today’s operating systems and the various methods in which sophisticated users conceal or eliminate evidence have created a challenging situation for investigators: there is more data than ever before, but at the same time there exist many more options for users to conceal or destroy evidence. The New Technology File System (NTFS) developed by Microsoft has indirectly replaced FAT32 as the primary file system being used on Windows-based systems. NTFS provides extended reliability and security, as well as advanced metadata capabilities, allowing for better utilisation of large quantities of data.

### A. Background of NTFS Forensics

NTFS doesn’t simply remove the pointers to files when you delete them. Instead, it has a complex series of operations in various system files to manage the deallocation of files. The Master File Table (\$MFT) acts as a central relational

database for all files and directories present on the disk. When you use the “Shift+Delete” command to delete a file, the OS will update the MFT record of that file to indicate that it is “unallocated” and will update the \$Bitmap file to show that the clusters that were occupied by the file data are now available for use by new files. However, the metadata (or digital ‘DNA’) of the file will also remain in the MFT record until it is replaced by the creation of another file.

### B. Problem Statement

This study addresses the main research question about why metadata remnants may be sensitive to change over time and to what extent these metadata remnants will remain intact for an undetermined period of time. Various suspects in criminal cases will unsuccessfully try to securely delete files by deleting them directly from My Computer instead of the Recycle Bin, thereby permanently deleting all traces of their existence. Current traditional forensic tools include only automated carving techniques; however, because file systems can become fragmented or transaction logs can experience some level of corruption, the use of automated carving techniques in forensic investigations may not properly retrieve all metadata. Thus, there exists a need for a methodology that combines both automated parsing techniques with manual deep architectural analysis in order to ensure that all possible investigative leads are fully exhausted or ruled out due to limitations posed by data recovery tools.

### C. Research Objectives and Scope

The main goal of this study is to determine whether analyzing NTFS system files (e.g., \$MFT, \$LogFile, and \$Bitmap) can provide support for the reconstruction of timelines for deleted files. In particular, the study includes:

- **Architecture Analysis:** Document and describe the internal structure of the \$MFT records (specifically, the 1024-byte records) and the attribute headers of each of the associated records (e.g., \$STANDARD\_INFORMATION, \$FILE\_NAME).
- **Experimental Deletion Scenarios:** Controlled removal of data from a 10GB NTFS partition to obtain forensic images in the .E01 format (standard in the digital forensic industry).

- **Tool Validation:** The effectiveness of the ingest modules (both automated and command-line using The Sleuth Kit (TSK)) will be compared based on the percentage of successful recoveries.

By providing a well-defined framework for the analysis of metadata, this research will make a significant contribution to the field of cyber security and digital forensics and validate the approach to the recovery of evidence from NTFS environments.

## II. LITERATURE REVIEW

There's been quite a shift in how we do computer forensics through NTFS since FATs went out of style, and this paragraph ties together all of the latest research on metadata persistence (or lack thereof), transaction log analysis (and recovery) and the flaws with any automated forensics recovery systems.

### A. Metadata Persistence and MFT Evolution

The initial research conducted on NTFS primarily focused on exploring the structure of the Master File Table. Researchers found out that, although \$MFT is a file index, it is, in fact, a complex database. More recent research into the subject has expanded this understanding of the files being stored by looking at the "residual" data that is left in the 1024-byte MFT records after a file has been flagged as unallocated. The research has shown that even after the operating system has flagged the MFT record with "0x00" meaning deleted, the existing attributes are usually still intact until that record is reused by another file entry.

### B. Journaling and Transaction Log Analysis

The \$LogFile and \$UsnJrnl have an extensive amount of research available on them. Many researchers have asserted that the \$LogFile is a resource that investigators should be taking advantage of more in their investigations. Previous research has found that the metadata transactions are stored within a circular buffer, providing investigators with a "time-machine" effect of being able to see the extent of a file system before files were deleted. There is a documented gap in research on how the \$LogFile (\$LogFile is represented as a log file for the NTFS file system) behaves when a volume has a high degree of fragmentation; this could cause log entries to be interleaved and/or corrupted.

### C. Comparative Analysis of Forensic Tooling

Similar to many other articles indexed in both the IEEE and Scopus databases regarding digital forensic investigations, this article discusses the efficacy of automated tools compared to manual analysis. Automated tools such as Autopsy and FTK Imager have the capability to construct timelines quickly; however, researchers have identified the limitations of the carving algorithms used in these tools. By using The Sleuth Kit (TSK) to examine files manually, many times, investigators can find metadata residue (for example, the MAC time stamp) that will not be identified by automated ingest modules that process data at high speeds.

### D. Impact of Modern Storage Technologies

Current studies have started addressing how the logical file system NTFS functions at a higher level than hardware-based functions like the SSD TRIM command. The findings indicate that for solid-state devices (SSDs), the time available to recover metadata after data is deleted is greatly reduced in comparison to traditional magnetic hard drives because the TRIM command proactively removes deallocated clusters in order to sustain improved write speeds.

### E. Synthesis and Identified Research Gap

Even though there has been much research done, there has been little substantive information documenting how to reconstruct deleted files sourced from corrupted transaction logs in fragmented systems. Prior studies have primarily been done while assuming the health of a particular file system. This study provides a methodology by which to identify correlations between residues located within the \$MFT and \$Bitmap based on incomplete primary journaling data.

## III. NTFS ARCHITECTURE

The forensic capabilities of NTFS can be established through a detailed study of the file system's internal organizational structure. NTFS is a complete filesystem and, therefore, manages everything as a file; whether that is the file being managed by the filesystem, or the filesystem itself (i.e., the file used to manage the filesystem).

### A. The Volume Boot Record (VBR)

The investigation starts with the Volume Boot Record (VBR), which is located within the first sector of the partition. The VBR stores a critical part of metadata, referred to as the BIOS Parameter Block (BPB), which contains the following information:

- Sector Size - 512-byte sectors (in general).
- Cluster Size - 4KB (8 sectors) (in general).
- The MFT Logical Cluster Number (LCN) - A pointer that tells the computer where the Master File Table begins.

### B. The Master File Table (\$MFT)

The Master File Table (\$MFT) is the backbone of the NTFS file system because every file gets an entry in the table, even the \$MFT itself. Each record has a fixed size of 1024 bytes (1 KB). The first part of each record is the record header, which contains both the "FILE" signature and an update sequence number array. Next to the header is a 2-byte allocation flags value found at the 0x16 offset. A value of 0x01 means that the file has been allocated, where a value of 0x00 means that it has been deleted. We have utilized this flag to seek out remnants of 'unallocated' records within our study.

### C. Critical Attribute Analysis

The NTFS file system includes various types of attributes, and two of the most important when reconstructing data for a forensic purpose are:

- **\$STANDARD\_INFORMATION (Type 0x10)** - This attribute provides “basic” metadata for a file, which includes the 8-byte timestamp values for the creation, modified (changed), updated (changed), and accessed timestamp (known as the MACM timestamps).
- **\$FILE\_NAME (Type 0x30)** - The second important attribute provides a Unicode representation of the file’s name, as well as provides a second set of MACM timestamps. When a file is deleted from the NTFS volume, the residue of the file name is kept in this attribute, which can be recovered for forensic reconstruction of deleted files.

#### D. Ancillary System Files: \$Bitmap and \$LogFile

Deallocation of resources is completed in a manner that relies on both the MFT and additional system files for synchronisation.

- **\$Bitmap:** This file contains a bitmap of each cluster on the disk volume. We employ bitmasking logic in this file to establish whether or not any of the clusters associated with a deleted file have been either written over or remain as “stale” clusters.
- **\$LogFile:** The log file serves as a circular buffer to store the transaction log. The log file can contain “redo” and “undo” operations for the deletion of a record, even if the MFT record has only been partly written over. Thus, the log file is intended to be used to reconstruct the events of the deletion process at a later time.

### IV. METHODOLOGY AND EXPERIMENTAL DESIGN

The credibility of forensic research relies heavily on isolation and reproducibility within an investigative setting. This section outlines the layout of the virtualized laboratory, the parameters for generating data, and the protocols used for acquisitions that ensure evidence cannot be altered.

#### A. Experimental Environment and Hardware

In order to mitigate host-system contamination and create a “clean room” environment, a virtualized laboratory was created.

- **Host System:** The analysis was conducted on a MacBook Pro with an M4 Max chip (16-core CPU and 40-core GPU), which has 48GB of Unified RAM.
- **Virtualization Layer:** VMware & VirtualBox have been used to host a virtual machine instance of Windows 10. Target Partition: A separate 10GB NTFS partition was created specifically to store all forensic artifacts collected from the target system.

#### B. Forensic Software Suite

To be able to compare the results of an automated parsing approach to those of a manual architectural analysis, we chose a multi-layered toolset:

- **Acquisition:** FTK Imager was set up for bit-stream disk imaging actions, therefore producing E01 Forensic Images.

- **Automated Analysis:** Autopsy 4.21 was used for its ingest modules with a focus on MFT Parsing and Timeline Analysis modules.
- **Low-Level Parsing:** The Sleuth Kit Command-line Tools (TSK) had a use for manual examination of raw metadata. Specifically, we will use fls and istat.
- **Hex Analysis:** HxD Hex Editor was also used to visually analyze the headers’ raw data from the \$LogFile and the header from an \$MFT record.

#### C. Controlled Deletion Scenarios

To replicate actual anti-forensics methods, the study tested two main types of deletion:

- Standard deletion, which consisted of moving a file to the recycle bin and then deleting it from the recycle bin, caused the deallocation of clusters from the \$Bitmap file.
- Permanent deletion (using Shift + Delete) by bypassing the recycle bin caused the original MFT record to immediately be flagged in the unallocated record.

In addition to analyzing test deletion methods, many types (DOCX, JPG, PDF) of random test data were utilized to determine how file type (signature and size) impacts the existence of residual metadata.

#### D. Evidence Validation and Acquisition

After removing the data from the 10GB partition, it was unmounted from the Windows operating system to ensure that new data would not be written over the space allocated for the previously deleted files. The unmounted partition was then used to create a bit-stream image of the 10GB partition that allowed for verification and preservation of its forensic integrity through the use of SHA-256 hash values prior to analysis; thereby confirming that this evidence was dynamic and immutable during any subsequent analyses.

### V. RESULTS AND FORENSIC ANALYSIS

This analysis phase was looking at how automated tools correlated with manual verification of NTFS Metadata Residual files. The results of the testing were taken from the 10GB Forensic Image file (.E01) created after a controlled “Shift + Delete” action.

#### A. Experimental Environment and Hardware

The MFT Parser was able to identify 15 records as being unallocated (0x00) after the image was imported into Autopsy. Upon checking the deleted MFT records with The Sleuth Kit’s istat tool, all of the \$STANDARD\_INFORMATION attribute headers were present. 80% recovery of deleted file metadata occurred prior to any overwriting on the portion of the partition where the records resided. \$FILE\_NAME attribute (Type 0x30) had retained the original file name in Unicode for all instances where the corresponding MFT record had not been reused.

TABLE I  
METADATA RECOVERY SUCCESS BY ARTIFACT TYPE

| Artifact Target | Metadata Captured         | Success Rate | Forensic Value      |
|-----------------|---------------------------|--------------|---------------------|
| SMFT            | Filename, MAC Timestamps  | 90%          | Primary Evidence    |
| \$LogFile       | Transaction History       | 65%          | Temporal Sequencing |
| \$Bitmap        | Cluster Allocation Status | 95%          | Proving Deletion    |
| \$Standard Info | Creation/Access Dates     | 80%          | Timeline Integrity  |

### B. Comparative Effectiveness of System Files

A primary objective was to evaluate the recovery potential of different NTFS system files. The results are summarized in Table I below:

### C. Manual Hex Analysis vs. Automated Parsing

While the Use of Autopsy provided a detailed visual timeline of events, it did not determine what particular residuals had been written to the \$LogFile when there was some level of corruption. For example, the use of HxD enabled us to perform a manual search and identify “undo” versions from the transaction log, which ultimately enabled us to reconstruct a file’s existence when the MFT record for that file had been corrupted. Also, upon analyzing the circular buffer contained in the \$LogFile, we were able to locate evidence of the “Shift+Delete” event, which had originally been classified as a standard system event by the automated timeline tools. Furthermore, checking the \$Bitmap revealed that even though the cluster was marked as 0 (available), the original file header (e.g. PDF 0x25 0x50 0x44 0x46) would still exist in that cluster; thus confirming that a deallocation does not equal data destruction.

### D. Data Visualization of Recovery Success

The findings indicated that Metadata contained in the MFT (as resident data) from smaller files (DOCX) has a higher percentage of remaining in the MFT in relation to the larger files (PDF), where metadata is stored in relation to cluster pointers and therefore more at risk of being overwritten.

## VI. DISCUSSION

NTFS’ file system’s forensic resilience was widely validated here by experimental data. Due to the differentiated loss of data, regardless of whether it was due to accidental deletion, corruption, or physical failure, we were able to demonstrate that there is at least an 80-percent recovery rate for metadata items. Interpretation of these data establishes that there is a complex relationship between the efficiency of both automation processes and the precision of manual processing.

### A. Analysis of Tool Performance and Limitations

One major observation in the Analysis Phase was that there were inconsistencies between methods of automated ingest (modules) versus manual hex parsing. While Autopsy provides a solid view of a visual timeline of activity, it also falls short in its ability to detect leftover fragmented metadata entries (residues) within the \$LogFile.

- **Automated Parsing:** These types of tools usually use standard signatures. Therefore, if a portion of the transaction log has been corrupted or has a “dirty” status, the automated parser will not be able to identify valid records.
- **Manual Verification:** Using The Sleuth Kit (TSK) to parse the raw 1024-byte record structures of the MFT allowed the team to see hidden attributes that were not indexed in the high-level GUI.

### B. The “Metadata Residue” Phenomenon

According to the findings of this research study, it has become evident that “permanent” deletions on NTFS are inaccurate descriptors. Although clusters will be shown as deallocated through the \$Bitmap, the metadata still resides within the associated MFT in a “stale” condition. This retention of cluster information largely results from the lazily written nature of the file system which focuses on performance rather than providing data sanitization immediately. For Investigators - the presence of timestamp residues in \$STANDARD\_INFORMATION and filename residues in \$FILE\_NAME can be used to establish a presumption of intent or “guilty knowledge” regardless of whether or not the actual contents of the file(s) can be recovered.

### C. Anti-Forensics and Modern Hardware Challenges

One of the major points of discussion in the recent literature and in our planning for future work is the impact of SSD TRIM commands.

- **Hardware Evolution:** Metadata was not automatically disposed of after a change; on traditional HDDs, it remains in existence until it has been physically overwritten.
- **The Impact of TRIM:** On modern SSDs, there is an active command sent to the controller telling it that it must delete any deallocated or unused area of the drive which is likely to greatly reduce the thirty-five (35) per cent maximum recovery rate that we have seen in a virtualised (HDD emulated) environment.

### D. Reliability of Transaction Logs

The \$LogFile was not just a valuable artifact, it was also very volatile. The circular buffer characteristics of the log mean that, if an active system is present after deletion of the log, its most recently accessed entries will be the first to be deleted from the log because the ‘undo/redos’ are stored at the beginning of a circular log (circular buffer). Our recovery effort had great success, mainly as a result of the fact that a 10GB partition was immediately unmounted right after the deleted events occurred.

## FUTURE WORK

While this study shows that NTFS metadata can be recovered at a high rate in a virtual HDD environment, there remain many emerging challenges related to digital forensic investigations that still need additional research.

- **The effects of SSD TRIM operations.** During the planning phases of this project, it was noted that with the transition from magnetic HDDs to SSDs comes the use of a TRIM command. On an HDD, data remains until it is overwritten. On an SSD, TRIM will proactively wipe out deallocated clusters to help maintain write efficiency into the future. Addressing the aforementioned challenge will involve performing experiments with HDD emulated virtual machines versus physical hardware-based SSDs in order to quantify the decrease of remnants of metadata.
- **The analysis of fragmented and corrupted volumes.** One of the gaps identified during the research literature review process was that little research has been done on the recovery of NTFS metadata from fragmented NTFS partitions where the \$LogFile is interleaved or partially corrupt. The next phase of this project will involve developing advanced “carving” algorithms that recover NTFS metadata based on bit stream patterns rather than the MFT structure.
- **Utilizing machine learning for automated parsing.** Due to the limitations found within current automated digital forensic tools, it may be possible to utilize machine learning to identify patterns in unallocated MFT records.

### CONCLUSION

The results of the study show that a great deal of material left behind after the so-called “permanent” deletion of files in an NTFS environment can be found in various metadata structures within the NTFS file system. Researchers analyzed the 12-week project through both the analysis of the low-level data (hex-level analysis) and high-level analyses (such as calendar analysis) and studied the history of evidence.

It was found that the following results were obtained as part of this study:

- **High Metadata Retention Rate:** 80% of critically relevant metadata, such as the original name of the file and its associated MAC timestamp, will still exist after performing a controlled “Shift + Delete” operation in relation to the \$MFT and \$LogFile.
- **Tools Have Disparate Capabilities:** Automated tools like Autopsy are useful for quickly developing timeline visualizations of evidence, but investigators need to conduct manual examinations of evidence to locate residual data that their ingest modules may not be able to detect using tools like The Sleuth Kit (TSK) and hex editors.
- **Methodological Validity:** The study employed bit-stream imaging and SHA-256 hash verification as methods to maintain the integrity of the experimental data, which met the requirements for a rigorous approach to digital forensic analysis.

The ongoing analysis of NTFS metadata by digital forensic investigators will provide a compelling means of establishing the integrity of digital evidence. While the evolution of anti-forensic tools will continue to pose challenges, there will be significant value in maintaining a firm understanding of

the internal structures of NTFS file system management files, specifically, the \$MFT, \$Bitmap, and \$LogFile.

### REFERENCES

- [1] Thompson, “Women and feminism in technical communication,” *J. Bus. Tech. Commun.*, vol. 13, no. 2, pp. 154–178, 1999.
- [2] M. S. MacNealy, *Strategies for Empirical Research in Writing*. Boston, MA: Allyn and Bacon, 1999.
- [3] J. H. Watt and S. A. van den Berg, *Research Methods for Communication Science*. Boston, MA: Allyn and Bacon, 1995.
- [4] S. Kleinmann, “The reciprocal relationship of workplace culture and review,” in *Writing in the Workplace: New Research Perspectives*, R. Spilka, Ed. Carbondale, IL: Southern Illinois University Press, 1993, pp. 56–70.
- [5] K. St. Amant, “Virtual office communication protocols: A system for managing international virtual teams,” in *Proc. IEEE Int. Professional Commun. Conf.*, 2005, pp. 703–717.
- [6] Structural Engineering Society–International. [Online]. Available: <http://www.seaint.org>.
- [7] M. Tohidi et al., “Getting the right design and the design right: Testing many is better than one,” in *Proc. ACM-SIGCHI Conf. on Human Factors in Computing Syst. (CHI’06)*, 2006, pp. 1243–1252.
- [8] B. Carrier, *File System Forensic Analysis*, Addison-Wesley Professional, 2005.
- [9] M. Russinovich, D. Solomon, and A. Ionescu, *Windows Internals*, Microsoft Press, 2012.
- [10] J. Sammons, *The Basics of Digital Forensics*, Syngress, 2014.
- [11] A. S. Lim, “A Forensic Analysis of the NTFS LogFile,” *Journal of Digital Forensics*, 2011.
- [12] NIST, “Computer Forensics Tool Testing Program (CFTT),” National Institute of Standards and Technology, 2023.
- [13] G. Richard III and V. Roussev, “Scalable Digital Forensics,” *Communications of the ACM*, 2021.
- [14] H. Carvey, *Windows Forensic Analysis Toolkit*, Syngress, 2014.