Efficient Recovery of Deleted Data and Metadata from NTFS and ReFS File

Ms. R.Kalaiyarasi(*Assistant Professor*), Arvind S
Dept. of Computer Science and Engineering
Sri Shakthi Institute of Engineering and Technology
Coimbatore, India
kalaiyarasiapcse@srishakthi.ac.in
arvinds22cse@srishakthi.ac.in

Dinesh P, Gavyaa A M
Dept. of Computer Science and Engineering Sri Shakthi Institute of Engineering and Technology
Coimbatore, India
dineshp22cse@srishakthi.ac.in
gavyaaam22cse@srishakthi.ac.in

Girija Nagarajan , Joselin J
Dept. of Computer Science and Engineering
Sri Shakthi Institute of Engineering and Technology
Coimbatore, India
girijanagarajan22cse@srishakthi.ac.in
joselinj22cse@srishakthi.ac.in

Abstract - Data loss resulting from accidental deletion, file system corruption, malware attacks, ransomware incidents, and unexpected system failures continues to present a significant challenge in modern Windows-based computing environments. As digital storage systems increasingly support critical personal, organizational, and enterprise data, the consequences of data loss have become more severe, affecting operational continuity, legal compliance, and forensic investigations. Although numerous data recovery solutions are currently available, the majority of these tools focus primarily on restoring raw file content and often neglect the recovery of associated metadata such as file names, timestamps, directory hierarchy, access permissions, ownership details, and security descriptors. Metadata plays a vital role in preserving contextual accuracy, forensic validity, and auditability of recovered data, particularly in investigative and compliance-driven scenarios.

This paper presents a comprehensive metadata-aware recovery framework specifically designed for Windows file systems, namely NTFS and ReFS. The proposed approach leverages file system–specific internal structures, including the Master File Table (MFT) in NTFS and copy-on-write metadata trees in ReFS, to recover deleted files along with their associated metadata. A unified metadata correlation and validation engine is employed to ensure consistency, accuracy, and forensic reliability of recovered records by cross-verifying multiple metadata attributes. The framework operates in a read-only manner to maintain forensic integrity and prevent evidence contamination.

Key Terms - NTFS, ReFS, Metadata Recovery, Data Recovery, Digital Forensics, Windows File Systems, Master File Table (MFT), Copy-on-Write, File System Metadata, Integrity Streams.

I. INTRODUCTION

The rapid expansion of digital information in recent decades has significantly increased the demand for reliable data storage, protection, and recovery mechanisms. Digital data now serves as a critical asset across various domains, including personal computing, corporate operations, healthcare systems, financial institutions, government agencies, and cloud service providers. Modern computing environments depend extensively on digital storage systems to manage sensitive information such as confidential documents, financial records, intellectual property, multimedia content, and mission-critical databases. Consequently, any loss of data can have severe implications, ranging from operational downtime to reputational damage and legal consequences.Windows operating systems are among the most widely used platforms globally and predominantly rely on NTFS and ReFS file systems to manage storage. NTFS has been the default file system for Windows for several decades, offering features such as journaling, access control, and detailed metadata management. ReFS, introduced as a next-generation file system, focuses on scalability, data integrity, and fault tolerance, particularly for enterprise and server environments. Despite incorporating advanced reliability and integrity mechanisms, both file systems are still susceptible to data loss incidents. Data loss can occur due to a wide range of factors, including accidental deletion by users, malware infections, ransomware attacks, logical corruption of file systems, software bugs, power failures, improper shutdowns, and hardware malfunctions. In many cases, users become aware of the importance of the lost data only after it has been deleted or corrupted, making recovery a critical requirement. As storage capacities grow and file systems become more complex, traditional recovery techniques face increasing limitations. content, it typically ignores file system metadata. As a result, recovered files often lack filenames, directory paths, timestamps, and access permissions. The absence of metadata significantly reduces the usefulness of recovered data, especially in digital forensic investigations where establishing timelines, ownership, and user activity essential.

A. Objective

The primary objective of this work is to design and implement a robust metadata-aware recovery framework capable of restoring deleted data along with its associated metadata from NTFS and ReFS file systems. The framework aims to reconstruct complete file context, including filenames, timestamps, directory hierarchy, ownership information, and access permissions, while maintaining forensic integrity throughout the recovery process. Additionally, the system seeks to improve recovery confidence by minimizing false positives and ensuring that recovered records are logically consistent and verifiable.

B. Scope

The scope of this research includes the logical analysis and recovery of deleted data and metadata from NTFS and ReFS file systems within Windows-based computing environments. The framework supports recovery across multiple file types, including text documents, images, audio files, and video files, and emphasizes forensic reliability and metadata validation. The study also includes a comparative analysis of recoverability characteristics between NTFS and ReFS. Recovery from physically damaged storage media and hardware-level failure scenarios are considered beyond the scope of this work.

II. RELATED WORK

Early research in file system forensics highlighted the importance of understanding internal file system structures to achieve accurate and reliable data recovery. Carrier's pioneering work on file system forensic analysis emphasized the role of metadata, journaling mechanisms, and allocation strategies in determining recoverability. His research provided foundational insights that influenced the design of many contemporary forensic and recovery tools used in NTFS based investigations. Fairbanks expanded on this work by examining NTFS deletion behavior and demonstrating how overwritten or partially corrupted MFT entries directly impact recovery accuracy. His studies revealed that even minor corruption within metadata structures can significantly reduce the effectiveness of traditional recovery tools. These findings underscored the need for recovery approaches that prioritize metadata analysis rather than relying solely on content-based techniques.File carving techniques introduced by Richard and Roussev focus primarily on recovering file content using file signature analysis. Although these methods can successfully reconstruct certain file types, they completely disregard metadata reconstruction, resulting in incomplete forensic evidence. Research on ReFS conducted by Suiche provided a detailed analysis of copy- on-write mechanisms and integrity streams, highlighting how these features improve data consistency while simultaneously complicating post-deletion recovery. More recent studies advocate metadata-driven recovery frameworks, demonstrating that correlating multiple metadata attributes significantly improves recovery accuracy and forensic dependability.

III. SYSTEM ARCHITECTURE

The proposed system adopts a modular, layered architecture designed to support both NTFS and ReFS file systems while maintaining strict forensic integrity. The architecture ensures separation of concerns, scalability, and extensibility, enabling future enhancements without disrupting core functionality. All system components operate in a strictly read- only mode to prevent evidence contamination and accidental data modification.

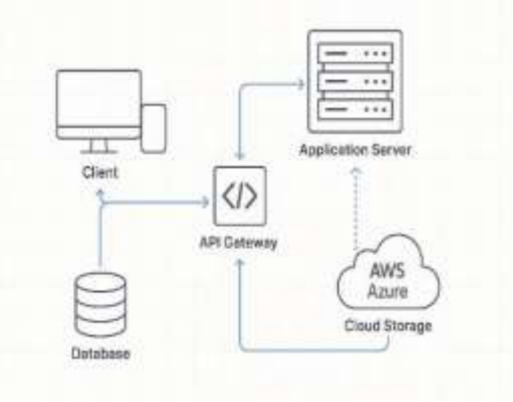

Fig.1. Overall System Architecture of the Metadata-Aware Recovery Framework

A. Disk Interface Module

This module provides secure, read-only access to disk images or storage devices. It abstracts low-level sector operations and ensures that no write operations occur during analysis.

B. File System Identification Module

This module analyses volume metadata to determine whether the underlying file system is NTFS or ReFS. Accurate identification is essential for selecting the appropriate metadata parsing logic.

C. Metadata Parsing Engine

The metadata parsing engine extracts and interprets file system–specific metadata structures. NTFS recovery relies on MFT parsing, while ReFS recovery involves metadata tree traversal and integrity stream analysis.

D. Recovery Decision Module

This module determines whether extracted metadata records are recoverable by applying heuristic and rule-based checks to filter incomplete or corrupted records

.

E. Validation and Consistency Module

Recovered metadata is validated for timestamp consistency, allocation status, and directory hierarchy. This step reduces false positives and improves forensic reliability.

F. Forensic Logging Module

All recovery actions and metadata attributes are logged in detail to support auditability and legal admissibility.

IV. METHODOLOGY

A. Existing System

Existing recovery tools rely heavily on raw disk scanning and signature-based file carving. These tools fail to reconstruct metadata accurately and offer limited support for ReFS.

B. Drawbacks of Existing System

- Loss of filenames, timestamps, and

directory hierarchy
- High false positive rates
- Limited ReFS support
- Poor forensic reliability

C. Proposed System

The proposed system performs logical file system analysis and reconstructs metadata using structure-aware parsing. Metadata is recovered before content, improving accuracy.

D. Advantages of Proposed System

- Preserves metadata and file context
- Supports NTFS and ReFS
- Reduces false positives
- Maintains forensic integrity

V. IMPLEMENTATION

A. HMARF Dashboard and Recovery Simulation Control

The HMARF dashboard acts as the central interface for configuring and initiating recovery simulations. Users can select the target file system type (NTFS, ReFS, or both) and specify the number of files involved in the simulation. This design allows comparative evaluation of recovery behavior across different file systems.

Once the parameters are selected, the recovery simulation can be initiated through a single control action. A progress indicator provides real-time feedback on the simulation status, ensuring transparency and usability.

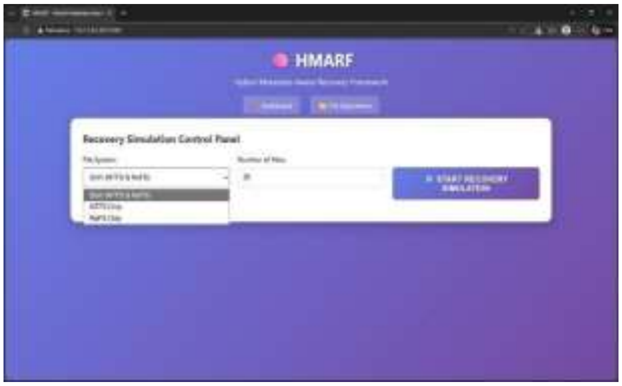

Fig.2. HMARF Dashboard and Recovery Simulation Control Panel

B. File Operations and Directory Monitoring Module

The File Operations module enables interaction with real directories in the system. Users can specify a directory path, list files within the directory, and perform add or delete operations. Each file operation is automatically recorded in the operation logs along with associated metadata such as file name, size, timestamp, and checksum.

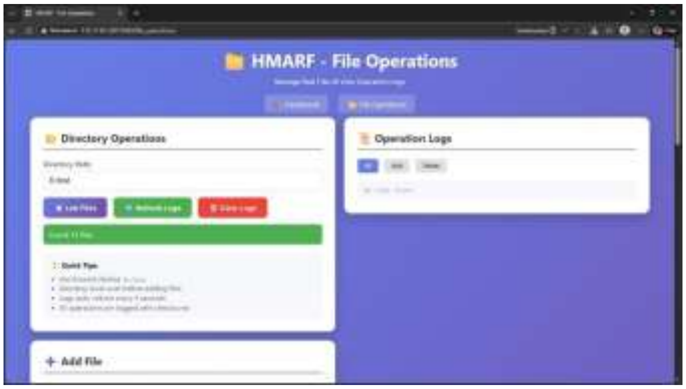

Fig.3. File Operations and Directory Monitoring Interface

C. Operation Logs and Metadata Tracking

Operation logs form the backbone of the metadata-aware recovery approach. Every add and delete operation is recorded with precise metadata attributes. These logs serve as the primary source of information during the recovery phase, allowing the system to reconstruct deleted files even when raw data recovery is incomplete. The logs are categorized based on operation type, enabling users to filter and analyse file system activities efficiently. This approach closely aligns with forensic principles, where historical activity reconstruction is critical.

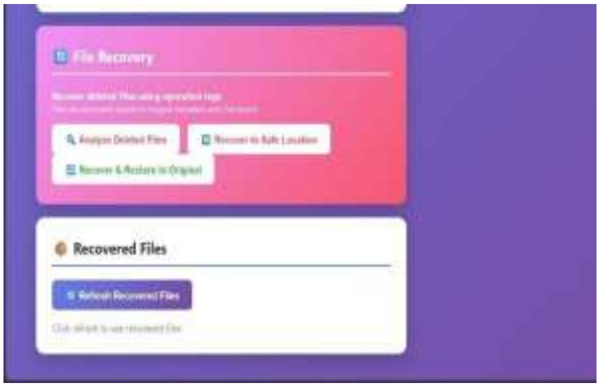
File Recovery
Recover deleted files using operation logs
Analyze Deleted Files
Recover to Safe Location
Recover & Restore to Original
Recovered Files
Refresh Recovered Files
Click refresh to see recovered files

Fig.4. Operation Logs with Metadata and Action Classification

D. Recovery Simulation Control Panel

The Recovery Simulation Control Panel allows users to configure and initiate the metadata-aware recovery process in the HMARF system. Users can select the target file system (NTFS, ReFS, or both) and specify the number of files involved in the simulation. Once initiated, the system executes the recovery process and displays a real-time progress indicator. Upon completion, the progress bar confirms successful execution of the recovery simulation.

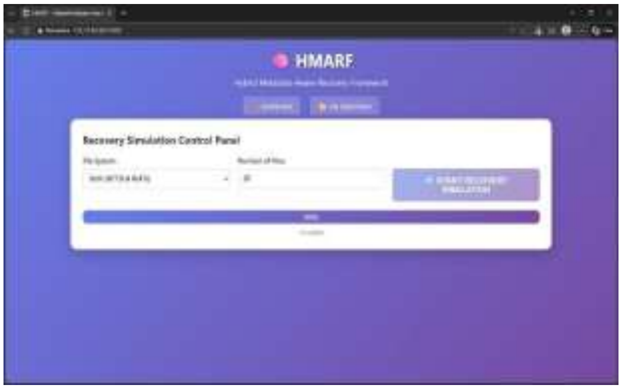

Fig 5. HMARF Recovery Simulation Control Panel

E. Recovered Files Display and Validation

This interface presents the detailed list of files recovered from the ReFS file system after completion of the recovery process. Each entry includes the file name, file size, recovery status, and a download option for verification. Files labeled as Verified indicate that both file content and metadata have been successfully restored, whereas files marked as Partial represent cases where metadata recovery was successful but content recovery was incomplete. This view helps in assessing the effectiveness and reliability of the recovery framework.

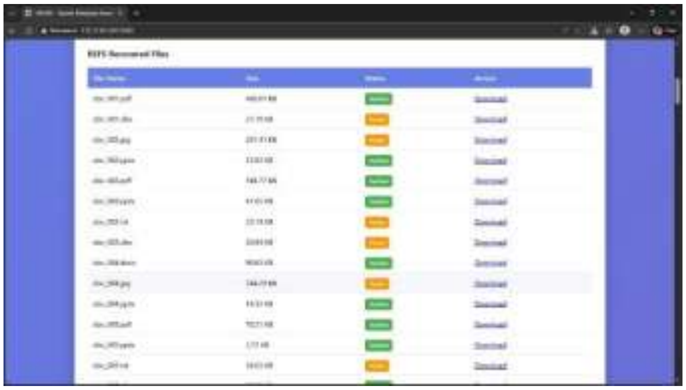

Fig.6. ReFS Recovered Files List Showing File Details, Recovery Status, and Download Options

VI. EXPERIMENTAL SETUP

A. Test Environment

Experiments were conducted on Windows systems using NTFS and ReFS partitions. Multiple file types were used for evaluation.

B. Dataset Description

Datasets included text documents, images, audio files, and video files of varying sizes.

VII. RESULTS AND DISCUSSION

A. Results

Experimental results indicate that NTFS provides higher metadata recovery rates due to persistent MFT entries. ReFS demonstrated stronger integrity protection but lower recoverability.

B. Discussion

The results highlight a fundamental trade- off between recoverability and integrity. NTFS favors forensic analysis, while ReFS prioritizes reliability and resilience.

VIII. CONCLUSION

This paper presented a comprehensive metadata-aware recovery framework for NTFS and ReFS file systems. By emphasizing logical metadata interpretation and validation, the framework significantly improves recovery accuracy and forensic reliability in Windows environments.

IX. FUTURE WORK

Future enhancements include improving ReFS recovery strategies, supporting additional file systems, integrating automated forensic reporting, and optimizing performance for large-scale storage environments.

REFERENCES

[1]M. Alazab and S. Venkatraman,” Digital Forensics in Windows-Based File Systems,” *International Journal of Information Security*, Vol. 15, No. 3, 2016.

[2] N. Beebe and J. Clark, “Challenges in Forensic Analysis of Modern File Systems,” *Journal of Digital Forensics, Security and Law*, Vol. 4, No. 2, 2009.

[3] R. Behl and S. Behl, “Metadata-Based File Recovery in Windows File Systems,” *International Journal of Advanced Computer Science and Applications*, Vol. 8, No. 6, 2017.

[4] A. Case and G. G. Richard, “Windows File System Artifact Analysis,” *Journal of Digital Investigation*, Vol. 7, No. 2, 2010.

[5] B. Carrier, “File System Forensic Analysis: Concepts and Techniques,” *Digital Investigation*, Vol. 2, No. 3, 2005.

[6] E. Casey and B. Carrier, “Digital Forensic Analysis of File System Artifacts,” *Journal of Computer Security*, Vol. 12, No. 3, 2004.

[7] K. Fairbanks, “Recovery of Deleted Files from NTFS File Systems,” *Digital Investigation*, Vol. 4, No. 2, 2007.

[8] S. Garfinkel, “Automated File System Analysis for Digital Forensics,” *IEEE Computer*, Vol. 46, No. 3, 2013.

[9] S. Garfinkel and A. McCarrin, “Integrity and Reliability in Modern File Systems,” *IEEE Security and Privacy*, Vol. 15, No. 4, 2017.

[10] P. Gladyshev, “Formal Approaches to File System Forensics,” *Digital Investigation*,vol.6.No. 3, 2009.

[11] T. Haigh,“Evolution of File Systems and Storage Reliability,” *ACM Computing Surveys*, Vol. 44, No. 3, 2012.

[12] J. Kim and H. Lee, “Performance Comparison of NTFS and ReFS File Systems,” *International Journal of Computer Systems Science*, Vol. 11, No. 2, 2019.

[13] A. Kumar and R. Singh, “Comparative Study of NTFS and ReFS File Systems,” *International Journal of Computer Engineering and Technology*, Vol. 10, No. 1, 2019.

[14] V. Kumar and P. Sharma, “Analysis of Data Integrity and Recovery in NTFS and ReFS,” *International Journal of Computer Applications*, Vol. 181, No. 12, 2020.

[15] O. Mason, “Design and Reliability Features of the Resilient File System (ReFS),” *International Journal of Storage Systems*, Vol. 9, No. 1, 2012.

[16] J. Metz, “Analysis of NTFS Metadata Structures for Digital Forensics,” *International Journal of Digital Evidence*, Vol. 7, No. 1, 2010.

[17] S. Patel and D. Shah, “Metadata Management in Windows File Systems,” *International Journal of Advanced Research in Computer Science*, Vol. 9, No. 4, 2018.

[18] D. Richard and G. Roussev, “Scalable File Carving Techniques for Modern File Systems,” *Digital Investigation*, Vol. 2, No. 1, 2005.

[19] M. Russinovich, D. Solomon, and A. Ionescu, “NTFS Architecture and Metadata Management,” *Microsoft Systems Journal*, Vol. 18, No. 4, 2012.

[20] S. R. Soltan, G. H. Gulak, and V. C. M. Leung, “A Survey on File System Forensics,” *IEEE International Conference on Communications*, Vol. 1, No. 1, 2018.