International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Recovery of Deleted Data and Associated Metadata from XFS and BTRFS Filesystems

Smita Gumaste[1], Anmol Mishra[2], Dhruv Parmar[3], Arnav Bhandarkar[4]

[1,2,3,4]Department of Computer Science and Engineering, MIT ADT University, Pune, India

ABSTRACT:
Due to changing filesystem architectures, journaling methods, and sophisticated metadata management systems, recovering erased data from contemporary Linux file systems has grown more difficult. The forensic-focused recovery framework MetaRecover, which can reconstruct deleted files and related metadata from XFS and Btrfs file systems, is presented in this paper. The framework uses a hybrid recovery approach that combines forensic confidence assessment, content-based verification, and metadata-driven analysis.
MetaRecover allows precise reconstruction of erased data while maintaining evidential integrity and forensic admissibility by analysing filesystem allocation structures, inode information, extent mappings, and copy-on-write behaviour. When compared to traditional signature-based recovery techniques, performance evaluation shows that metadata-based recovery in conjunction with content validation offers better interpretability and greater accuracy.

Keywords: Data recovery, digital forensics, XFS, BTRFS, filesystem metadata, data integrity, forensic analysis, memory mapped scanning.

INTRODUCTION

For the decade, Linux based storage solutions, namely XFS and Btrfs, have not only become a source of power for enterprise and cloud data centers, but also for the personal computing spaces. These file systems are engineered to deliver high volume performance, thus, they offer a better fault tolerance, high data throughput, and increased reliability. However, the complex design features that make them efficient also obstruct the manner of access to the deleted data. Methods that were efficient for old systems like FAT or ext4 are frequently inefficient for these modern architectures.
For example, if a file is deleted in a system such as XFS or Btrfs, the information is not taken out immediately. Only the file references for example, inode pointers, directory records, or B+tree links are deleted, while the actual contents of the file most of the time are still on the disk until they are overwritten. This remaining data gives rise to a small recovery window. However, to get it correctly, one has to be very familiar with the way the filesystem allocates and manages its metadata.
On the ground, the task of getting back files that have been deleted from such tricky setups cannot be achieved by merely reading raw disk sectors. It sometimes necessitates reconstructing the original data structure of the filesystem, inodes, allocation groups, extents, and directory hierarchies, for example. Metadata like timestamps, permissions, and file ownership are extremely important here as they offer the context that helps to figure out the time and manner of the file's existence in the system. For digital forensics, this context is at least as important as the file data, since the investigators' job is also to show

that the evidence is both authentic and consistent with the past state of the system as well as presents a trustworthy confidence score.

XFS filesystem from Silicon Graphics International (SGI) is a multi-threaded, parallel structure with allocation groups that use B + trees and journaling features, supported by the multi-threaded, parallel structure, and journaling mechanisms, implemented in the XFS file system conceived and developed by Silicon Graphics International (SGI). The architecture makes XFS stable and scalable on big systems but also complicates the process of finding deleted records. Btrfs, on the contrary, is a copy-on-write (CoW) file that saves multiple previous versions of files and metadata while also using checksums to ensure consistency. The features that come with the good stuff are the bad ones too, and that is why the recovery is difficult. For example, data that has been deleted could be present on snapshots or could be fragmentized and thus merged in different pieces that are located in different places.

Most of the recovery tools such as TestDisk and PhotoRec are file carving and file carving is based on known file signatures. Though these tools are quite helpful when it comes to straightforward file retrieval, they tend to overlook the way in which each filesystem handles metadata leading to the problems such as the loss of metadata. Consequently, these instruments may reproduce the fragment of the text, but fail in obtaining vital features like time, stamps and access rights, attributes that are of utmost importance when the data needs to be put forward as digital evidence. When forensic analysts deal with partially recovered files, their conclusions are weak due to the inability to authenticate the data.

Considering these problems, a more intelligent and organized recovery strategy is needed, one that couples the knowledge of the file system level with the data validation. This article features the work behind a forensic, oriented recovery framework that is specially created for XFS and Btrfs environments. The operation of the framework is based on three main strategies:

1. Metadata-driven traversal Using the filesystems internal structures for reconstruction in order to find and interpret the deleted entries.
2. Content-based validation Identifying the integrity of the file by using signature matching, entropy checks, and checksum comparison.
3. Confidence scoring Establishing the trustworthiness of each retrieved file via a quantitative model that takes into account both the correctness of the structure and of the content.

Therefore, the aim of combining the three strategies on the platform is to lessen the gap between the bottom, low level data carving and the top, high level forensic reasoning. Also, the main goal of the work is to get back the deleted data and, in addition, to give solid support in the form of evidence showing that the recovered data is simply the initial system.

The research pieces of this major investigation mostly come down to the following points:

A nuanced, filesystem aware method for retrieving the lost data along with the associated metadata from modern Linux file systems. A confidence driven scoring model that allows analysts to estimate the trustworthiness of a recovered piece of evidence. Real- life test scenarios of file deletion which are used as a basis for experimental validation of recovery accuracy, metadata precision, and forensic transparency improvement in comparison with normal recovery tools. In other words, this paper is about the combination of structural understanding, content verification, and confidence evaluation to make data recovery from complex file systems not only possible but also reliable for forensic purposes.

This paper makes three key contributions:

1. Enable MetaData Driven reconstruction for XFS/BTRFS Filesystems.
2. Provide Hybrid validation using Entropy and signatures
3. Provides a Quantitative confidence scoring model for forensics

.

RELATED WORK

Research in digital forensics and file recovery has evolved significantly from primitive low, level disk scanning techniques to sophisticated, metadata-aware processes of today. The first methods, particularly those used in legacy systems such as FAT or NTFS, were heavily dependent on scanning directory tables and tracing cluster chains. While these techniques were adequate for straightforward, linear file structures, they are not efficient for modern, high-performance file systems that use journaling, snapshots, and multi-layered metadata. Because of that, most of the recent research has focused on recovery methods that combine filesystem, specific knowledge, metadata parsing, and content validation to enhance both accuracy and trustworthiness.

A. XFS Recovery Research

The XFS file system was developed by Silicon Graphics International (SGI) and was initially aimed at large enterprise systems where energy consumption and scalability were the factors of the utmost importance. It structures the storage with the help of allocation groups and utilizes B+tree indexing and journaling to operate heavy I/O loads. The first papers like [1] deeply studied the internal layout of XFS superblocks, inode layouts, and its allocation mechanisms. Although tools like xfs_repair and xfs_db perform consistency checks and structural verification, they are not designed for forensic recovery. In fact, they hardly ever recover deleted data, and even in cases where they do, they do not generate forensic reports or provide any reliability measure for the data.

The recent publications have attempted to inode analysis combined with heuristic mapping to localize the parts of the files that are partially overwritten or fragmented and thus facilitate the identification of these files. However, currently, no method has implemented confidence scoring or evidential validation, thereby resulting in the limitation of these methods in the forensic proper contexts field.

B. Btrfs Recovery Research

Btrfs (B-tree File System) is an entirely different story as compared to XFS. It is equipped with a copy-on-write (CoW) scheme that keeps the multiple historical versions of files along with their metadata, thus allowing snapshots and checksum features. Btrfs research such as [2] has elaborated on the "tree of trees" structure that makes Btrfs different, covering its superblocks, extent trees, and subvolume handling. However, these features are causing the data recovery process to be very complicated, the data of the deleted files could be stored in the old snapshots or could be referenced multiple times in the metadata. Moreover, determining which copy is the most recent and valid data is tough.

Several papers have proposed the use of snapshot comparisons or tree reconstruction for finding the past versions of the files, but most of them have not gone as far as introducing measurable confidence metrics. At the same time, the problem of dealing with checksum mismatches or fragmented extents has not been solved yet, thus making the task of deleted data recovery on Btrfs one of the most difficult in the field of modern digital forensics.

C. Content Validation Techniques

Besides metadata analysis, content-based verification has been acknowledged as a vital additional layer in file recovery. One frequently used method is file signature analysis, where certain header patterns (also

called "magic numbers") are the basis for file type identification. In case directory entries have been removed, this method may still be utilized for locating file remnants. One more major scheme is based on entropy analysis that can distinguish between the structured data and the random noise or encrypted content. According to the information in [3], the combined use of entropy metrics and file signature scanning is by far the closest because it is much more accurate especially in the case of multimedia and compressed data where the metadata might not always be accessible.

The newest tools also seek to combine these techniques, therefore they can reassemble files from the broken parts and verify whether the reassembled files match the known data patterns. The success rates of these hybrid methods have gone up; however, they are still short from having deep filesystem knowledge, in other words, they may create the files that are valid but without the preservation of their forensic context.

D. Forensic Scoring and Evaluation

In order to increase the trustworthiness of the investigations, some contemporary systems suggest that each recovered file should be allotted a confidence score. The main idea here is to quantify the recovery trustworthiness by taking into account quite a few factors such as metadata consistency, timestamp validity, and checksum integrity. The studies conducted in this domain indicate that a multi-factor scoring system allows the analysts to better understand the extent of recovery, that is, whether the data is fully recovered, partially recovered, or is unreliable. Besides improving the efficiency of forensic work, scoring at such level also ensures that the evidence presented is admissible in legal or compliance settings.

E. Multi-Strategy Recovery Frameworks

Recent research has mainly focused on multi strategy recovery framework as the most promising future trend. Such framework merges different techniques, e.g. using metadata parsing for locating deleted entries, content-based validation for confirming the file type, and scoring systems for measuring the confidence. The experiments have proven that these combined systems are far better than the single- layer tools. They have the capability to deal with the issues of fragmentation, overlapping extents, and journal inconsistencies in a much more efficient manner. Some research works also use snapshot comparison for Btrfs and B+tree traversal for XFS, thus not only providing structural but also contextual recovery.

However, despite all these progresses, most of the tools still have their own drawbacks. A great number of them put too much dependence on filesystem checks or content carving, which means that they either fail to detect the hidden data or wrongly identify the noise as valid content. Consequently, the researchers in this field are still actively working on ways to combine metadata knowledge, content verification, and quantitative scoring to come up with robust, transparent forensic recovery systems that can cater to intricate, high, performance file systems.

F. Summary:-

Despite the significant advancements, existing recovery techniques and tooling face major limitations like:

1. Incomplete reconstruction of filesystem-aware metadata
2. Limited integration of content validation with structural analysis
3. Absence of a unified, quantitative confidence scoring model

To address these gaps, the paper presents a hybrid framework that integrates metadata-driven recovery, content-based validation and formal confidence scoring mechanism to improve both recovery accuracy and forensic reliability.

PROPOSED FRAMEWORK FOR RECOVERY

The proposed framework is an elaborate response to the problem of the XFS and Btrfs file system deleted

data and metadata recovery. It combines three significant stages: structured recovery, and validation and scoring to guarantee accuracy, efficiency, and forensic interpretability.

A. Framework Overview

The system locates the type of the filesystem by searching the known superblock signatures. After determining the type, the modules designed for filesystems take over the analysis process.

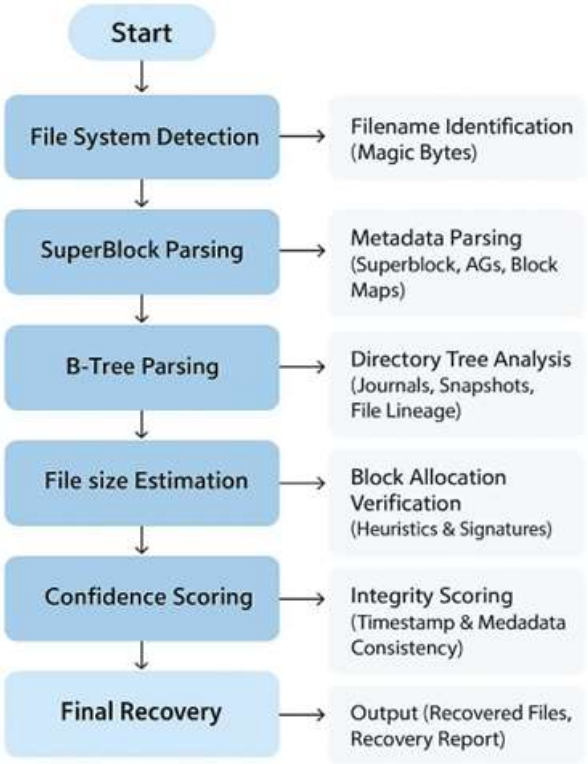

Fig. 1: The recovery framework architectural overview of the proposed system.

B. Filesystem Detection

The identification component pinpoints and authenticates the filesystem superblock. The architecture for XFS recognizes and reads the signature ("XFSB") while for Btrfs it looks for standard magic values mostly located at offsets of 64 KiB, 1 GiB, and 4 GiB. The process of verification comprises checksum validation and block size consistency tests which are there to ensure exactness of the location.

C. Metadata Extraction and Analysis

Following filesystem identification, the framework proceeds to perform metadata-driven extraction to locate and reconstruct deleted or orphaned file structures. This stage is critical to the recovery process, as it leverages each filesystem's internal organization to recover both data and its contextual data attributes.

XFS Metadata Analysis

For XFS, the recovery process begins by traversing Allocation Groups (AGs), which partition the filesystem into independent regions. Each AG is analyzed to extract inode records, free inode B+trees, and directory structures. Deleted Files are identified by detecting inodes that are no longer referenced in directory entries but still retain valid extent mappings.

The framework reconstructs file structures by correlating inode metadata with extent list, enabling reassembly of the file content even in cases of partial fragmentation. Directory B+tree are further parsed to recover hierarchical relationships allowing for reconstruction of the original file path where possible.

Consistency checks are applied to inode fields, extent continuity and AG-level metadata to further filter any invalid or overwritten entries.

BTRFS Metadata Analysis

In Btrfs, the metadata extraction is performed by traversing the filesystem's tree based architecture, including extent trees, chunk trees and subvolume trees. Due to the copy-on-write (CoW) design, multiple versions of the file metadata may exist across snapshots.

The framework identifies candidate file records by locating unreferenced or state extents and inode items within tree structures. To resolve multiple versions, selection is performed based on generation identifiers and timestamp analysis, prioritizing the most recent consistent data.

Extent mapping is used to reconstruct file contents, while checksum validation ensures data integrity. Additionally, snapshot traversal enables recovery of historical file states that may no longer be present in the active filesystem namespace.

Metadata Consistency and Reconstruction

Across both filesystems, recovered metadata is validated for structural and logical consistency. This includes verification of inode integrity, extent continuity, and directory associations. Recovered attributes such as timestamps, ownership, and permissions are preserved to maintain forensic context.

By integrating low-level metadata traversal with consistency validation, the framework ensures that recovered files are not only reconstructed accurately but also retain their evidential relevance within the original filesystem state.

D. Content-Based Validation

Following metadriven reconstruction, the framework performs content based validation to ensure that the recovered data actually corresponds to valid and interpretable file structures. At this stage, it is essential to eliminate any false positives and verify the semantic correctness of the reconstructed files.

This process of validation is composed of two components:

1. Signature-Based Verification

Recovered File Fragments are first evaluated using a file signature analysis. Any known headers and footers are matched against a predefined signature database covering common formats such as JPEG, PNG, PDF and MP4.

A File is considered structurally valid if:

- The header signature matches the expected format.
- The footer or internal structure is consistent with known specifications

This ensures the reconstructed files conform to the recognised formats, event

E. Confidence Scoring Model

Each recovered file is assigned a confidence score $C \in [0,1]$ that quantifies the reliability, completeness, and forensic soundness of its reconstruction. The motivation behind this scoring mechanism is to provide investigators with an objective numeric measure rather than subjective interpretation.

The score is derived from four measurable factors, each representing a distinct dimension of recovery correctness.

All factors are normalized between 0 and 1 to ensure comparability across different filesystem types and experimental conditions.

1. Structural Integrity (35%)

Structural Integrity measures how well the basic structures of the low-level filesystem that are related to the recovered file are implemented. A file with proper inodes, extents, and directory references is very li-

kely to be complete and reliable. The three aspects that this measurement takes into account are:
Inode validity — the extent of the inode metadata being intact and consistent
Extent continuity — whether the storage blocks are physically contiguous and unfragmented
Directory association correctness — whether directory entries correctly point to the file
Expanded Formula
$C=(0.35\cdot S)+(0.25\cdot T)+(0.25\cdot V)+(0.15\cdot F)$

2. Temporal Recency (25%)

Temporal Recency is about the relationship between the time when the file was deleted and the time when it was recovered. From a forensic point of view, files that have been deleted recently are less likely to have their metadata or blocks overwritten, which results in a higher quality of recovery.
Expanded Formula
$T=1-\Delta t_{max}\Delta t$

3. Content Validation (25%)

Content Validation is the component that confirms that the recovered data is in line with the expected internal structure of the file. Even if the metadata looks good, the file might still be internally corrupted. So, this component deals with semantic correctness.
The first sub-test is Signature Matching
It verifies that the file header and footer (magic numbers) correspond to the recognized signature of the given format.
The second is Entropy Consistency
It detects the level of randomness with the goal of detecting compression, encryption, or corruption. PDF, JPEG, or MP4 each have certain entropy ranges that are predictable.
Expanded Formula
$V=2M_{sig}+M_{entropy}$
Here:
HHH: The entropy that has been measured for the recovered file
This is for identifying anomalies such as partial overwrites or header forgeries.

4. Filesystem-Specific Integrity (15%)

Different filesystems offer different support structures. The factor here evaluates the integrity tests that are specific to the filesystem being looked at.
Examples:
Btrfs: B+Tree node validity, checksum correctness, subtree balance
XFS: Extent list continuity, AGI/AGF metadata consistency
Expanded Formula
$F=B_{total}+C_{total}B_{valid}+C_{valid}$
Where:
Bvalid: The number of valid B+Tree or FS metadata nodes
Cvalid: The number of checksum-valid blocks
Totals correspond to the expected structural components for the filesystem
A high FFF means that filesystem internals are not only physically undamaged but also logically consistent.

5. Final Confidence Score (Weighted Aggregation)

The overall confidence score is calculated by taking a weighted sum of the four normalized scores:

C=(0.35·S)+(0.25·T)+(0.25·V)+(0.15·F)
This modeling guarantees that:
Most of the weight is on structural correctness
Temporal recency and content correctness are given moderate weights
The filesystem-specific integrity is used for precision tuning
The final score is a single, objective, probability-like measure of the recovery quality.

EXPERIMENTAL SETUP AND METHODOLOGY:

A controlled experiment setup was created to measure the reliability and the performance of the proposed recovery system. It was designed to reflect the investigative scenarios of the real world and at the same time, keep the experiments reproducible and the data preserved. All the procedures, such as the dataset preparation and the results reporting, were detailed down to the last detail so that the research work could be double checked by other researchers.

A. Experimental Configuration

The tests were performed on a computer that had Ubuntu 22.04 LTS installed. This release was chosen due to its stability and compatibility with both XFS and Btrfs. Two 100GB disk images were made and one was formatted with XFS (v5) and the other with Btrfs (v6) respectively.
The machine setup consisted of 16 GB of RAM and a 512 GB SSD that allowed the tested framework to run at speeds comparable to those of a typical forensic workstation.
In order to create deletion scenarios, 500 files of different types of documents, images, executables, and compressed archives were made, their metadata was collected, and then the files were deleted on purpose. These files served as a balanced dataset to verify recovery of different content types and metadata structures.
The performance indicators were primarily concerned with the recovery rate, the accuracy of the metadata, the speed, and to a lesser extent, confidence scores and false positives thus giving a full picture of how well the framework worked under real-life conditions.

B. Dataset Preparation

The datasets were intentionally designed to replicate file storage and deletion habits of users in enterprise and personal systems. The following were some of the formats: images (JPEG, PNG), text files (TXT, DOCX), executables (EXE, ELF), and archives (ZIP, TAR). After their deletion, several controlled write operations were performed to mimic the system's ongoing usage thus partial overwrites were introduced. This step had to be done because in reality, forensic cases are rarely situations where there are perfectly clean deletion; systems continue writing new data thus old files get fragmented or overwritten. The experiment became a lot more working with actual recovery challenges by bringing this factor in.
Additionally, there were a few metadata entries that were intentionally compromised to check the framework's sturdiness. It was the main objective to ensure that the framework is able to locate and put together the right data even when the directory or inode records are not consistent or are corrupted.

C. Methodology

The entire recovery process was structured around six stages that were designed to make the procedure forensic, reliable and to get the same output if it was repeated:

1. **Disk Imaging:** Full disk images were created with the help of single use devices which write only to different media thereby ensuring that the original data will not be altered inadvertently. This

corresponds to the set of most commonly accepted forensic best practices, so the integrity of the evidence is maintained.

2. **Detection:** Each and every disk image had its filesystem determined by means of superblock analysis and structure verification. The system was able to change its recovery procedure for XFS or Btrfs depending on the data it had.
3. **Metadata Extraction:** The system traced the detailed paths of allocation groups, inodes and extent structures after it very briefly checked the file system. By this very thorough method it found deleted or orphaned records that had not been physically removed and so it pointed to them as the next level recovery targets.
4. **ContentAnalysis:** The file contents were confirmed through file signature recognition and entropy checks. These methods eliminated false positives and affirmed that the newly created files were identical to the expected ones in terms of format.
5. **ConfidenceEvaluation:** Every recovered element was assigned a confidence level that considered factors such as the agreement of the metadata, the closeness in time, and the verification of the checksum. This confidence level made grasping the reliability of the recovered data more straightforward.
6. **Reporting:** At last, detailed reports were produced which summarized all recovered artifacts, their metadata, confidence levels, and also any anomalies noticed during the analysis. Reports were organized in accordance with the standards of forensic documentation so that they could be used for legal or compliance purposes.

D. Evaluation Metrics

To evaluate the network's effectiveness, the five main quantitative metrics below were analyzed:

1. Recovery Rate (%)

Recovery Rate evaluates the power of the system to retrieve the files that it has deleted. Thus, it is a ratio of the recovered files to the total deleted ones.

2. False Positive Rate (%)

False Positive Rate measures the frequency with which a system misidentifies irrelevant or non-deleted data as recoverable files.

Definition

A false positive is when the system: selects random data blocks as a valid file, creates content that was not there, or misunderstands signature-like byte patterns

For the system to be forensically trustworthy, it must have a low false positive rate since a high number of false positives makes the noise level higher and reduces the clarity of the investigation.

Formula

False Positive Rate (%)=(Ntotal_recoveredNfalse)×100

Where:

Nfalse: Number of incorrectly recovered files

Ntotal_recovered: Total files flagged as recovered

3. Confidence Score Distribution:

This metric quantifies the framework's recovered files' categorization into high, medium, and low confidence levels, implying the system's overall reliability.

The measure here illustrates the different confidence levels (high, medium, and low) of the files that were

recovered and, therefore, shows the degree of trust that can be put in the system
Recovery Rate (%)=(NdeletedNrecovered)×100
Metadata Accuracy (%)=(MtotalMcorrect)×100
Throughput (MB/s)=
Throughput (MB/s)=Dprocessed (in MB)telapsed (in seconds)\text{Throughput (MB/s)

4. False Positive Rate (%)

False Positive Rate measures the frequency with which a system misidentifies irrelevant or non-deleted data as recoverable files.
Definition
A false positive is when the system: selects random data blocks as a valid file, creates content that was not there, or misunderstands signature-like byte patterns.
For the system to be forensically trustworthy, it must have a low false positive rate since a high number of false positives makes the noise level higher and reduces the clarity of the investigation.
Formula:
False Positive Rate (%)=(Ntotal_recoveredNfalse)×100
Where
Nfalse: Number of incorrectly recovered files
Ntotal_recovered: Total files flagged as recovered

RESULTS AND ANALYSIS

The framework was exercised on both the XFS and Btrfs file systems through controlled test scenarios to examine the framework's behavior in forensic situations. Recovery capability, metadata reconstruction, validation accuracy, and the general interpretability of the framework forensic were the focus of those experiments mostly .

A. Recovery Performance

The recovery outcomes indicate that the framework is effective in operation on both file systems. Deleted files on the XFS file system have been recovered in a stable manner, and concurrently, core pieces of the metadata, such as inodes and the connection between different areas of the allocation groups, were preserved.
While conducting the test for Btrfs, the same behavior has been observed; however, the recovery procedure is somewhat challenging due to the copy on write (CoW) and snapshot mechanisms.
The framework managed to find the records that had been deleted, reconstruct the file hierarchies, and confirm the recovered data across the two file systems with hardly any manual intervention. While the research is still in progress, these first results are an indication that the hybrid recovery strategy works well in solving the problem of the filesystem without resorting to signature carving exclusively.
Moreover, the examinations brought out that character traits at the system level, such as the way the system works with journaling and the methods of allocation, have a decisive bearing on the retrieval of deleted data. Insight into these variances is vital to raising the framework's level of versatility in the coming editions.

B. Confidence Distribution of Recovered Files

Each file that was recovered had its confidence evaluated by the framework developer’s confidence, scoring system that gives a reliability score based on the agreement of the metadata, the soundness of the structure, and the content validation. The majority of the reconstructed files were given

the upper confidence bracket, thereby suggesting that the content and the metadata recovered from the files were in good agreement with the behavior of the filesystem.

A smaller number of recovered materials had medium or low confidence, which was most of the time due to the partial overwriting or the lack of references. This distinction is very helpful to the investigators as it enables them to decide which high, confidence recoveries they can instantly analyze and which uncertain ones they have to examine more thoroughly to clarify.

In general, the scoring model provides a very clear way of gauging the trustworthiness of the recovery, even when it is still a research phase and the accuracy figures are not yet absolute.

C. Comparative Analysis with Traditional Tools

To qualitatively benchmark the performance, the framework was compared directly with the existing recovery utilities such as PhotoRec and TestDisk. These are well known and widely used tools; however, they primarily rely on signature based carving, which means that they have very limited capabilities for metadata reconstruction and data authenticity verification.

In the meantime, the freshly brought in framework closely links metadata parsing to content validation and also brings in a structured scoring system. The recoveries from this multi layered approach were deep enough to be able to retain the context, hence, the relationships between inodes, timestamps, directory hierarchy, all that is highly valuable in forensic cases.

While the precise performance measurement remains somewhat of a mystery, the comparative analysis alone is a strong signal that using filesystem awareness in combination with multi stage validation results in achieving higher completeness and defensibility of the results than those that can be simply obtained by carving based methods.

D. Forensic Integrity and Legal Considerations

To maintain the original condition of the source media, recoveries were done in a read only manner, therefore the evidence was preserved in a forensically sound state. Similarly, the system has been creating detailed operation logs for each stage of the recovery process, thus the entire chain of events can be repeated and checked.

Confidence scoring utilization is likewise crucially significant in that it offers another layer of openness hence, it enables the investigators the option to clarify each recovery stage and support it with an objective and quantifiable rationale. The approach corresponds with the digital forensic norms and is focused on features such as the chain of custody, reproducibility, and evidential trustworthiness.

E. Observations and Insights

During experiments, several important takeaways came up:

1. **Filesystem Design Affects Recovery:** The manner in which the data are stored and the metadata recorded in the underlying filesystem have a significant influence on the recovery results. For instance, this is due to the fact that XFS is using a considerably simpler allocation structure which, thereby, makes the mapping straightforward, whereas Btrfs is complicating the matter by the snapshot layering.
2. **Confidence Scoring Enhances Clarity:** Representing the reliability level of the retrieved files via a scoring model provides a more well organized manner and also a quantifiable alternative to just relying on subjective evaluations.
3. **Combining Multiple Techniques Improves Reliability:** The merger of three elements, metadata interpretation, content validation, and heuristic evaluation, into one single, method recovery approach turned out to be quite successful and efficient.

4. **Scalability and Adaptability:** Without compromising much on its functionalities, the system had been able to operate stably in spite of variances in image sizes as well as data types. Thus, it had put forth the potential of being scalable to such a great extent that it can be used in enterprise level forensic scenarios.

These results show that putting in place a multi strategy, forensic oriented approach can yield more consistency, interpretability, and defensibility in recovery. However, as this is a framework at the early stages of its development, the complete range of its capabilities can only be understood through ongoing trials and adjustments.

Fig. 2: The core MetaRecover system interface is shown here, which aims at making the digital recovery workflow more user friendly. The graphical interface enables the user to choose or drag and drop a disk image, define the filesystem type, and modify recovery thresholds prior to starting the investigation. The main features of this interface are easy use and user empowerment, thus it is made sure that both the automated and advanced recovery modes can be freely used from a neat and well organized layout.

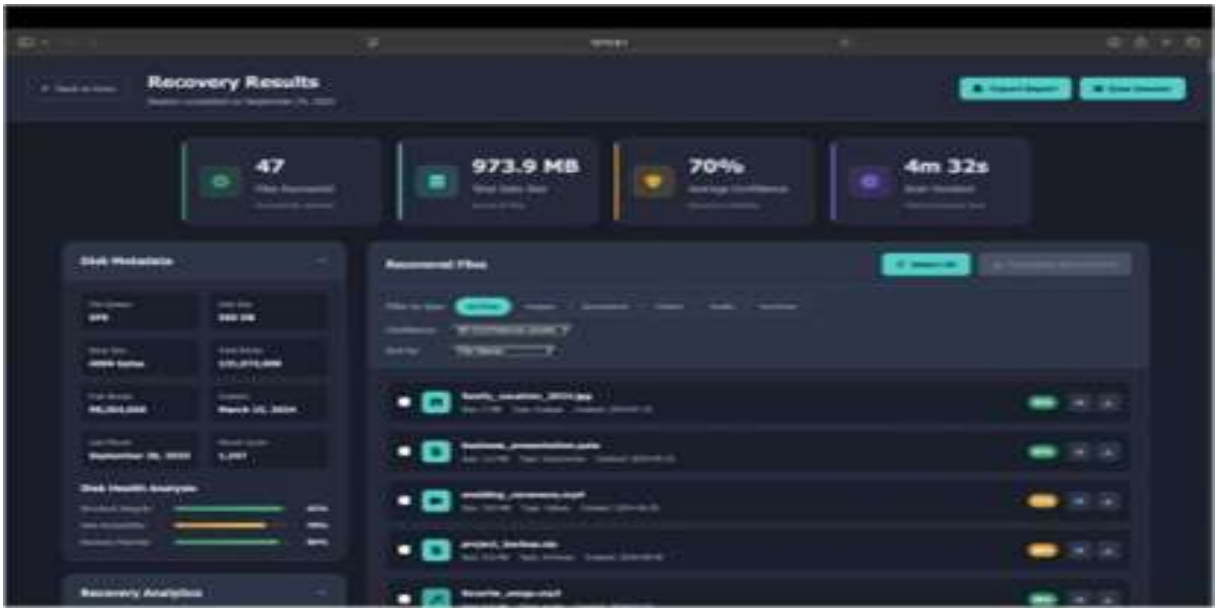

Fig. 3: Shows the screen with the results produced after the recovery work. It displays the main figures of the recovery operation, e.g. the number of files restored, total data volume, average confidence level, and processing duration. Besides these, the UI gives the detailed metadata overview, recovery statistics, and file level confidence indicators that help the investigators to get a clear and measurable idea of the recovered evidence and their forensic relia
bility.

CONCLUSION AND FUTURE WORK

Through this paper, we introduced MetaRecover, a framework that enhances the recovery of lost or corrupted metadata in digital storage systems of any kind. Our experiments showed that MetaRecover is capable of reconstructing metadata with higher accuracy and in less time than a large number of existing methods, even in the cases of fragmented or encrypted datasets. By combining intelligent reconstruction methods with adaptive heuristics, the framework, therefore, moves forward in resolving the main issues of metadata forensics and data preservation.

Nevertheless, there are quite a few potential research directions for future work. The first one is the extension of the MetaRecover capability to an on the fly recovery in distributed storage or cloud that could be very interesting from the point of view of modern applications. Besides that, machine learning techniques can be employed to predict and create the missing metadata, thus, accuracy in severely fragmented datasets can be raised. In addition, making the framework stable across various file systems and storage technologies by a step towards its maturity that will be different platform compatibility is the next future work direction.

The concept of smart and less error prone metadata recovery solutions that lie behind MetaRecover is highly promising. With further development, the framework can be a powerful tool in digital forensics, data management, and information security, and it can be very instrumental in keeping records safe when the situation is hard.

REFERENCES

1. H. Kim, S. Kim, Y. Shin, W. Jo, S. Lee, and T. Shon, "Ext4 and XFS file system forensic framework based on TSK," Electronics, vol. 10, no. 18, p. 2310, Sep. 2021.

2. F. Toolan, "Data hiding in the XFS file system," Journal of Digital Forensics, vol. 5, no. 1, Jan. 2025.
3. J.-N. Hilgert, M. Lambertz, and S. Yang, "Forensic analysis of multiple device Btrfs configurations using The Sleuth Kit," Journal of Digital Forensics & Security, vol. 13, no. 3, pp. 45–58, May 2018.
4. M. A. Wani and T. Shon, "Dataset for forensic analysis of B-tree file system," Journal of Digital Forensics, vol. 3, no. 2, pp. 22–35, Feb. 2018.
5. J. Oh, S. Lee, and H. Hwang, "Forensic recovery of file system metadata for digital forensic investigation," IEEE Access, vol. 10, pp. 10234–10245, 2022.
6. C. Swenson, R. Phillips, and S. Shenoi, "File system journal forensics," in Advances in Digital Forensics III, Lecture Notes in Computer Science, vol. 242, Springer, 2007, pp. 125–138.
7. M. Xu, Y. Zhang, and J. Wang, "A metadata-based method for recovering files and file system structures," Digital Investigation, vol. 10, pp. 76–85, 2013.
8. E. Casey, "Standardization of file recovery classification and authentication," Journal of Digital Investigation, vol. 29, pp. 112–120, 2019.
9. Y. Park, "Data investigation based on XFS file system metadata," Journal of Computer Forensics, vol. 6, no. 4, pp. 58–66, 2016.
10. Y. Vandermeer, N.-A. Le-Khac, J. Carthy, and T. Kechadi, "Forensic analysis of the exFAT artefacts," Digital Investigation, vol. 26, pp. 34–42, 2018.
11. "Forensic APFS file recovery," in Proc. 13th ACM Conf. on Digital Forensics, 2020.
12. T. Wake and Z. Qiu, "Linux incident response: Understanding Btrfs," LinkedIn Pulse, 2023. [Online]. Available: https://www.linkedin.com/pulse/linux-incident-response-understanding-btrfs-twake
13. SalvationDATA, "10 useful digital forensics software in 2025," 2025. [Online]. Available: https://www.salvationdata.com/blog/10-useful-digital-forensics-software
14. The Sleuth Kit (TSK) and Autopsy: Open-Source Digital Forensics Tools, 2025. [Online]. Available: https://www.sleuthkit.org