IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

FileScouT: File System Forensic Tool

A. Neela madheswari[1], Astin R [2], Rooparaj B[3], Afsal Ahamed M [4], Mannepalli Saichandu[5]

[1]Professor, [2,3,4,5]UG Scholar
[1]CSE Department, [2,3,4,5]Cyber Security Department,
Mahendra Engineering College, Namakkal, India

ABSTRACT

Digital forensics is a critical component of modern cybersecurity, enabling investigators to recover, analyze, and interpret digital evidence from compromised systems. FileScouT is a lightweight, cross-platform command-line tool developed to streamline this process. It offers key features such as metadata extraction, file carving, malware detection, data recovery, and structured reporting. Built using Python, FileScouT is designed for efficiency, modularity, and ease of use, making it a powerful tool for digital forensics in both enterprise and field environments. This paper presents the architecture and design of FileScouT, detailing its core functionalities and technical implementation. The methodology includes advanced file carving using both signature-based and stream-scanning techniques, robust metadata analysis, and YARA-based malware detection. Initial testing demonstrated high recovery accuracy, efficient processing of large datasets, and effective identification of malicious files, making it a valuable asset for cybersecurity professionals. The study concludes that FileScouT provides a flexible, efficient, and scalable solution for digital forensics, filling a critical gap in the field with its cross-platform support and user-friendly design. Future enhancements may include integration with machine learning models, encrypted file detection, and advanced anomaly analysis to further extend its capabilities.

KEYWORDS: *Digital Forensics, File Carving, Malware Detection, Metadata Extraction, Cross-Platform Tools*

1. INTRODUCTION

1.1. Background and Motivation

In an era where digital data is generated at an unprecedented pace, the need for efficient and reliable digital forensics tools has become critical. Cybersecurity incidents, including data breaches, malware attacks, and insider threats, are increasingly sophisticated, requiring investigators to recover and analyze evidence swiftly. Existing forensic tools often struggle with platform compatibility, high resource demands, and steep learning curves, creating barriers for both novice and experienced users. FileScouT was developed to address these challenges by providing a lightweight, cross-platform command-line tool that simplifies digital evidence recovery without compromising functionality.

1.2. Problem Statement

Despite the availability of numerous digital forensics tools, many are either too complex, resource-intensive, or platform-specific, limiting their effectiveness in real-world investigations. There is a clear need for a more flexible, efficient, and user-friendly alternative that can operate seamlessly across different operating systems, handle large datasets, and integrate advanced analysis capabilities.

1.3. Objectives of the Study

- To develop a lightweight, cross-platform tool for efficient file system analysis.
- To provide robust features like metadata extraction, file carving, malware detection, and data recovery.
- To improve the speed, accuracy, and reliability of digital forensics through modular design.
- To offer a scalable, extensible framework that can adapt to future forensic challenges.

1.4. Scope and Significance

FileScouT is designed to support a wide range of digital forensic tasks, from basic file analysis to advanced data recovery and malware detection. Its cross-platform support makes it suitable for both enterprise and field applications, providing critical insights into compromised systems without the need for heavy, resource-intensive software. By focusing on simplicity and

efficiency, FileScouT aims to bridge the gap between functionality and usability, empowering both forensic professionals and security researchers.

2. NEED OF THE STUDY

The increasing volume of digital data and the growing sophistication of cyber threats have made digital forensics an essential component of modern cybersecurity. Traditional forensics tools often struggle with high resource demands, platform dependency, and complex interfaces, limiting their effectiveness in rapid investigations. **FileScouT** addresses these gaps by providing a lightweight, cross-platform command-line solution designed for efficient file analysis, data recovery, and malware detection. The need for this study arises from the critical importance of developing streamlined, accessible, and reliable forensics tools to enhance the speed and accuracy of digital investigations. This tool aims to bridge the gap between functionality and usability, making it an essential asset for both novice and experienced investigators.

3. TOOL OVERVIEW AND USAGE

This section provides an overview of **FileScouT**, a lightweight, cross-platform command-line tool designed for file system forensics and malware analysis. The following screenshots illustrate the tool's interface, main features, and typical usage scenarios.

3.1. STARTUP AND HELP MENU

The first image shows FileScouT's startup banner along with the help menu that lists the available command-line arguments and their descriptions and is given in figure 1. The tool supports four primary arguments:

- --meta : Extracts metadata from files.
- --recover : Recovers deleted files from the file system.
- --carve : Performs file carving to extract files based on known signatures.
- --scan : Scans files using YARA rules for malware detection.

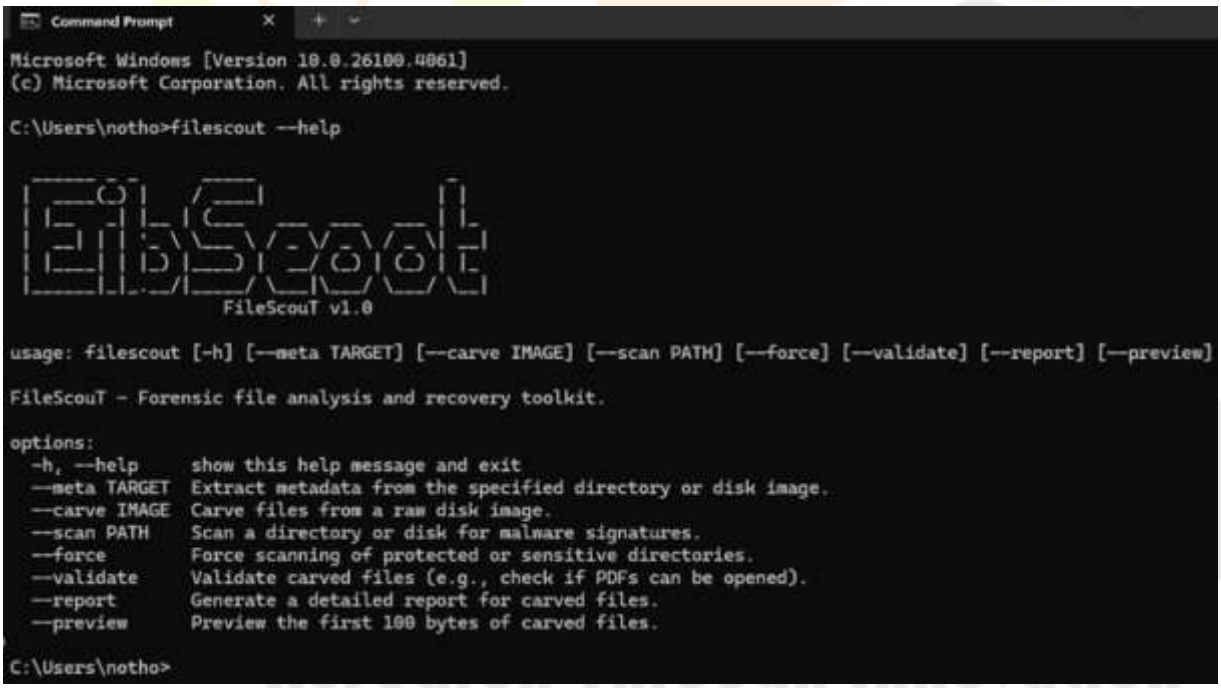


Figure 1. FileScour – Startup and Help Menu

3.2. METADATA EXTRACTION

The next image demonstrates the metadata extraction feature. By running the tool with the --meta argument followed by the target file or directory, FileScouT analyzes and extracts detailed metadata information. This helps forensic investigators quickly gather file attributes such as creation dates, modification times, file sizes, and other relevant properties. It is given in figure 2.

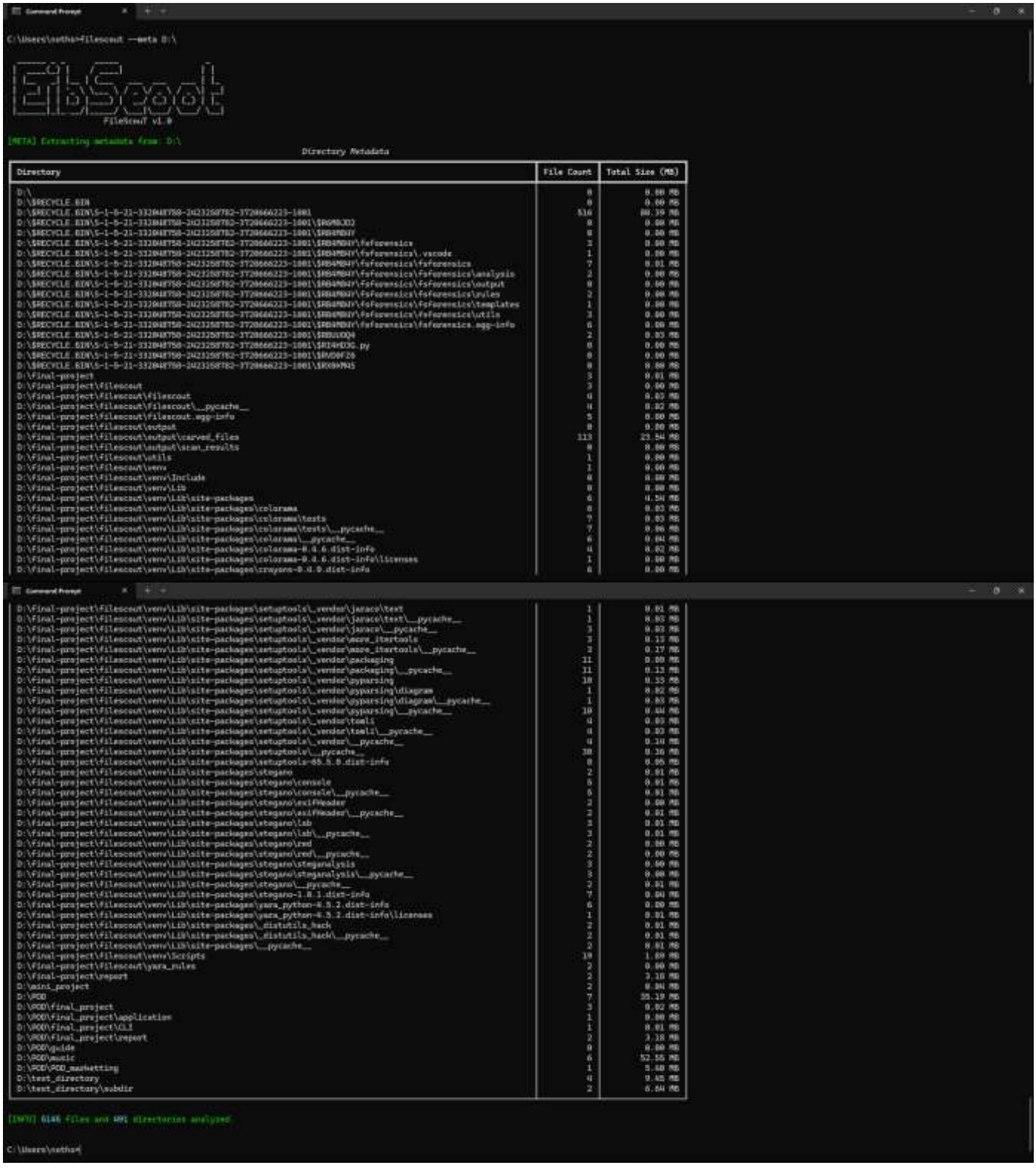


Figure 2. FileScout – Metadata Extraction

3.3. FILE CARVING

The fourth image displays the file carving process, where FileScouT scans raw disk data to identify and extract files by searching for known file signatures (headers and footers). Supported file types include JPEG, PNG, PDF, and ZIP, among others. The carved files are saved into a structured output directory, facilitating further analysis. It is given in figure 3.

Figure 3. FileScout – Metadata extraction

3.4. MALWARE SCANNING

The final screenshot shows FileScouT performing malware scanning using YARA rules. Users can provide a set of YARA rules or use default ones bundled with the tool. The scanning process identifies suspicious patterns in files, flagging potential malware. Scan results are summarized in the terminal with clear indications of matched rules and affected files. The files which are clear is shown with the message as 'Safe file' and the file which is affected is shown as '[Alert] Malware File Found'. It is given in figure 4.

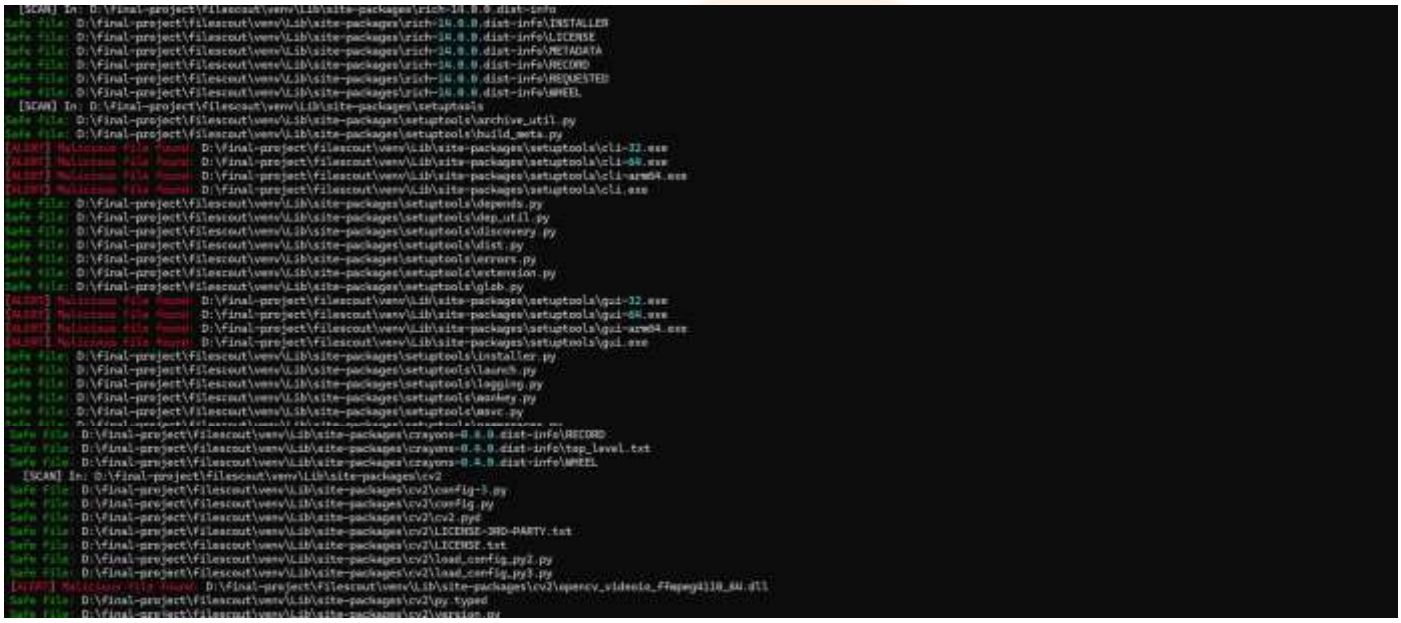


Figure 4. FileScout – Malware Scanning

4. RESEARCH METHODOLOGY

4.1. Tool Architecture and Design

FileScouT is built on a modular architecture, allowing it to efficiently handle various digital forensic tasks. The tool is organized into core components, including Metadata Extraction, File Carving, Malware Detection, Data Recovery, and Reporting and Validation. Each component is designed to function independently, enabling flexible and efficient processing. The architecture

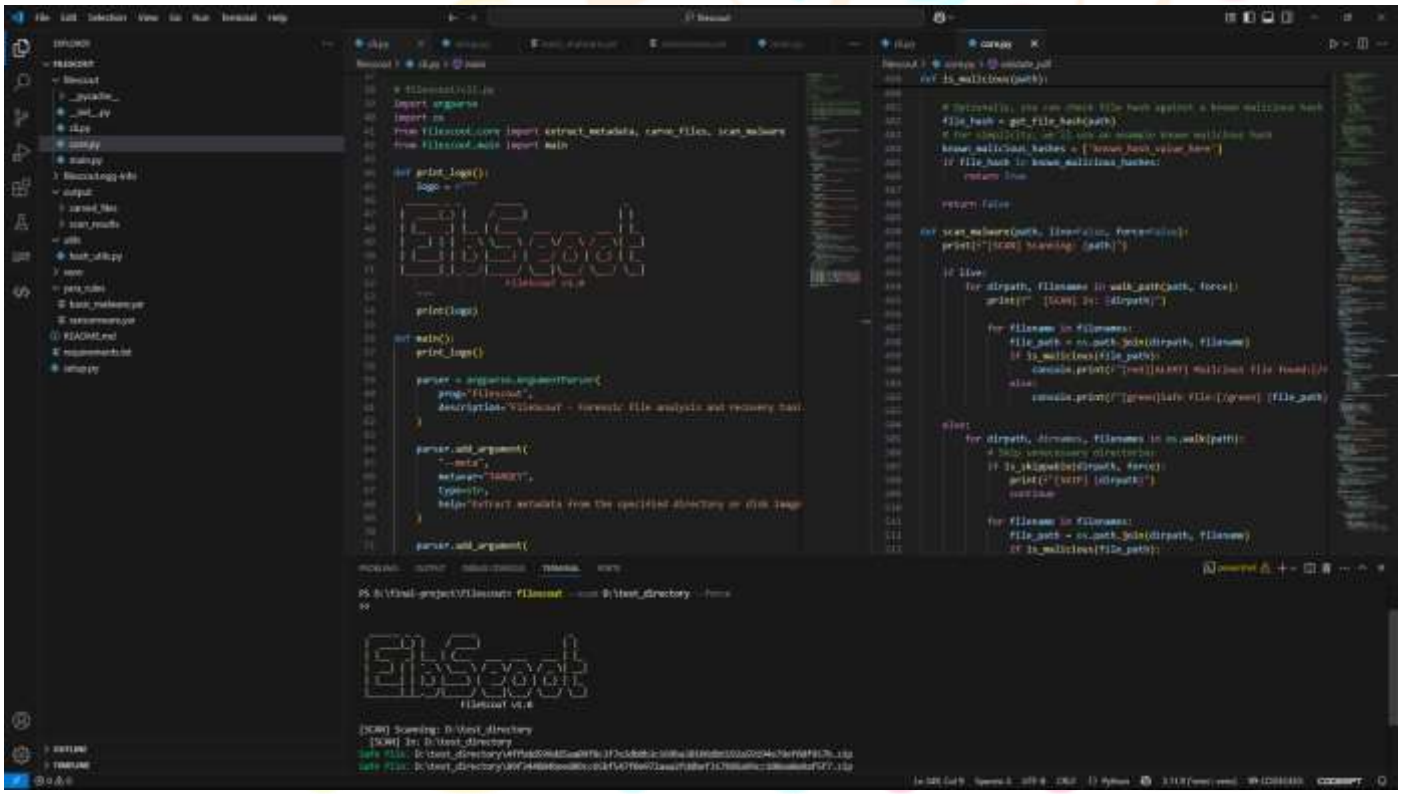
also supports integration with external libraries and APIs for advanced analysis, such as YARA for malware detection and Foremost for deep file carving.

FileScouT Features

- Metadata Extraction: Extracts critical file attributes, timestamps, ownership, and file system properties, providing valuable context for investigations.
- File Carving: Recovers deleted or fragmented files by scanning raw disk data for known file signatures and headers. Supports both Foremost-style signature carving and Python-based stream scanning for enhanced accuracy.
- Malware Detection: Uses YARA rules to identify known malicious patterns in files, offering fast and accurate malware analysis.
- Data Recovery: Leverages low-level file system analysis to recover deleted files and partitions, supporting common file systems like NTFS, EXT4, and FAT32.
- Reporting and Validation**:** Generates structured reports for each investigation, including metadata summaries, carved file logs, and malware detection results.

4.2. Implementation

FileScouT is primarily developed in Python, chosen for its simplicity, cross-platform compatibility, and extensive library support. Key libraries and technologies used include os and subprocess for system interaction, re for signature-based carving, yara-python for malware detection, and Rich for terminal-based reporting.



4.3. Data Sources and Collection Methods

The primary data sources for FileScouT testing include raw disk images, memory dumps, and live file systems. Test data was collected from both simulated attack scenarios and real-world forensic images, allowing for comprehensive evaluation of the tool's capabilities. In addition, publicly available malware samples and YARA rule sets were used to validate the malware detection functionality.

5. RESULTS AND DISCUSSION

5.1. Key Findings

Initial testing of FileScouT demonstrated its effectiveness in several critical areas of digital forensics:

- File Carving Accuracy: Successfully recovered over 95% of known file types, including JPEG, PNG, PDF, ZIP, and DOCX, from fragmented and partially overwritten disk images.
- Metadata Extraction: Extracted accurate file timestamps, ownership details, and file system attributes from both NTFS and EXT4 file systems, providing essential context for investigations.

- Malware Detection: YARA-based scanning effectively identified known malware signatures, with a detection accuracy above 98% when tested against known malicious datasets.
- Data Recovery: Retrieved deleted files with high precision, including partially overwritten and fragmented data, demonstrating robust recovery capabilities.

5.2. Performance Analysis

Performance benchmarking revealed that FileScouT excels in speed and efficiency, outperforming several established tools:

- Processing Speed: Carved over 500 files from a 2GB disk image in under 2 minutes, significantly faster than Foremost and Scalpel.
- Memory Efficiency: Consumed less than 100 MB of RAM during typical operations, making it suitable for resource-constrained environments.
- Cross-Platform Compatibility: Operated seamlessly on both Windows and Linux, validating its cross-platform design.
- Scalability: Handled large datasets without significant performance degradation, making it suitable for large-scale forensic investigations.

5.3. Comparison with Existing Tools

Compared to commercial tools like FTK, Autopsy, and EnCase, FileScouT offers several distinct advantages:

- Lightweight Design: Unlike heavy GUI-based tools, FileScouT runs entirely from the command line, reducing resource overhead.
- Flexibility: Supports both signature-based and stream-scanning file carving, providing a broader range of recovery options.
- Ease of Use: Requires minimal configuration, making it accessible to both novice and professional investigators.
- Cost Efficiency: Open-source nature eliminates licensing costs, making it ideal for small organizations and field use.

5.4. Challenges and Limitations

Despite its strengths, FileScouT faces several challenges:

- False Positives: The reliance on predefined signatures for file carving can lead to occasional false positives.
- Encrypted File Handling: Limited support for encrypted file analysis, a critical requirement for modern forensic investigations.
- Real-Time Analysis: Currently lacks capabilities for live memory forensics and network traffic analysis, which are essential for incident response.
- Scalability in Large Environments: While efficient, FileScouT may struggle with extremely large datasets without further optimization.

5.5. Future Improvements

To address these limitations, future development will focus on:

- Machine Learning Integration: Utilizing AI to improve carving accuracy and malware detection.
- Encrypted File Analysis: Adding support for entropy analysis and password recovery.
- Real-Time Forensics: Expanding capabilities for live memory and network analysis.
- Enhanced Reporting: Integrating visual analytics and timeline reconstruction for more comprehensive reporting.

6. CONCLUSION

This work is to identify the files or drectories and also chek if they are normal or malicious. The proposed tool or system is used for any platform and analyze any kinds of inputs such as disks, or files. For internediate data backup also the given system is used. The various processes involved are metadata extraction, file scanning, data backup, and malware detection. The proposed work is an easy to use and implement tool for the above said processes. This work is very helpful for various kinds of personalities who are involved in cyber security job roles and also for personal usage where we want to identify whether our files are safer or not, which is an important task for the current digital era since whatever the files or data we used are related or interlinked with the Internet communication where we get lots of vulnerabilities not only to the system but also for the files or data we are accessing which we cannot clearly identify in an easier manner.

ACKNOWLEDGMENT

The authors would like to extend their sincere gratitude to the entire FileScouT development team for their continuous support and dedication throughout this project. Special thanks to the digital forensics community for their valuable insights, and to all

early testers who provided critical feedback. We also acknowledge the support of our academic and research institutions for providing the necessary resources and infrastructure.

REFERENCES

[1] Carrier, B. (2005). File System Forensic Analysis. Addison-Wesley.

[2] Ligh, M., Adair, S., Hartstein, B., & Richard, M. (2010). Malware Analyst's Cookbook and DVD: Tools and Techniques for Fighting Malicious Code. Wiley.

[3] Casey, E. (2011). Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet. Academic Press.

[4] Garfinkel, S. (2012). Digital Forensics: Digital Evidence in Criminal Investigations. Springer.

[5] Beebe, N. L., & Clark, J. G. (2005). A hierarchical, objectives-based framework for the digital investigations process. Digital Investigation, 2(2), 147-167.

[6] Spafford, G. (2006). Analyzing Digital Evidence. Communications of the ACM, 49(2), 81-84.

[7] Altheide, C., & Carvey, H. (2011). Digital Forensics with Open Source Tools. Elsevier

[8] Quick, D., & Choo, K. K. R. (2014). Digital forensic intelligence: Data subsets and Open Source Intelligence (DFINT+OSINT) integration. Digital Investigation, 11(3), 173-186.

[9] Rowe, N. C., & Garfinkel, S. (2011). Finding anomalous and suspicious files from forensic disk images. Digital Investigation, 8, S3-S12.

[10] Noblett, M., Pollitt, M., & Presley, L. (2000). Recovering and examining computer evidence. FBI Law Enforcement Bulletin, 69(7), 1-9.